UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

Doble Grado en Ingeniería Informática y Matemáticas

TRABAJO FIN DE GRADO

HERRAMIENTA DE ESTIMACIÓN DE INDICADORES SOCIOECONÓMICOS EN BASE A TRAZAS DIGITALES DE REDES SOCIALES ONLINE

María Medina Pérez
Tutor: Esteban Moro Egido
Ponente: Pablo Castells Azpilicueta

JULIO 2015

HERRAMIENTA DE ESTIMACIÓN DE INDICADORES SOCIOECONÓMICOS EN BASE A TRAZAS DIGITALES DE REDES SOCIALES ONLINE

AUTOR: María Medina Pérez

TUTOR: Esteban Moro Egido

PONENTE: Pablo Castells Azpilicueta

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Julio de 2015

Resumen

Los indicadores socioeconómicos son utilizados para determinar el grado de desarrollo de las comunidades sociales y su análisis es crucial a la hora de diseñar políticas que mejoren la vida de sus ciudadanos. Sin embargo, su obtención es costosa puesto que requiere de la combinación de numerosas fuentes de datos o la realización de pormenorizadas encuestas a la población. Además, en países en vías de desarrollo es prácticamente imposible acceder a este tipo de estadísticas.

Con estos problemas en mente, los autores de un estudio publicado recientemente idearon una aproximación para inferir la tasa de desempleo a partir de una serie de indicadores extraídos de la red social Twitter. Definieron once variables relacionadas con el uso de Twitter, actividad diaria, contenido de los tweets e interacciones entre usuarios, y estudiaron hasta qué punto podía explicarse la variabilidad del paro en cada zona geográfica de España mediante estas variables, para un periodo de tiempo de siete meses.

Partiendo de las ideas de ese estudio, en este trabajo se ha creado un sistema que procese los datos de forma eficaz y calcule las variables que pueden aproximar el paro en distintas áreas geográficas predefinidas. Posteriormente, se ha utilizado esta herramienta para extraer los indicadores sociales en un periodo temporal de 2 años.

Estos datos se han utilizado para analizar cómo varía la capacidad de predecir la tasa de desempleo a lo largo de todo el periodo estudiado y finalmente se han diseñado y evaluado varios modelos predictivos que intentan resolver este problema. Los resultados son prometedores para tratarse de predicciones realizadas únicamente con datos provenientes de Twitter, pero no se acercan a la exactitud de otros modelos que utilizan indicadores de fuentes diferentes. Aun así, este estudio ha servido también para detectar varios aspectos que podrían refinarse para tratar de mejorar así estos modelos de predicción.

Palabras clave

Análisis de redes sociales, predicción de indicadores socioeconómicos, tasa de desempleo, extracción de variables de Twitter, grafo de movilidad, detección de comunidades.

Abstract

Socioeconomic indicators are used to determine the level of development in a certain social community, and analyzing them is crucial to designing policies that improve its citizens' life. However, obtaining them is costly since it requires combining many data sources or conducting detailed surveys to a part of the population. Furthermore, accessing these statistics is barely impossible in developing countries.

With all these issues in mind, authors of a recent research came up with an approach to infer the unemployment rate from a series of indicators extracted from the social network Twitter. They defined eleven variables related with Twitter usage, daily activity, tweet content and user's interactions, and studied how these variables could explain variability in unemployment rate in each geographical area in Spain, in a seven month period.

Based on the ideas from that paper, in this project we have created a system that processes data in an efficient way and calculates variables that can approximate unemployment in predefined geographic zones. Thereafter, this tool has been used to extract these social indicators for a period of two years.

This data has been used to determine changes in the variables' capability of forecasting the unemployment rate throughout the entire period and finally we have designed and tested several predictive models that try to address this problem. Results are promising given that predictions are made using input data exclusively from Twitter, but are far from the accuracy achieved by other models that use indicators from other sources. However, this investigation has also been useful to detect some aspects that could be enhanced to try to improve these predictive models.

Keywords

Social network analysis, forecasting of socioeconomic indicators, unemployment rate, extraction of Twitter variables, mobility graph in Spain, community detection.

Índice de contenido

Índice de figuras

Índice de tablas

1 Introducción

1.1 Motivación

Los indicadores socioeconómicos, como la tasa de desempleo, el Producto Interior Bruto (PIB), la tasa de escolarización o la esperanza de vida al nacer son utilizados frecuentemente para determinar el grado de desarrollo de una comunidad de habitantes. Esta información resulta de gran utilidad a la hora de desarrollar políticas que mejoren la calidad de vida de la comunidad, puesto que ayudan a definir objetivos a cumplir y a determinar qué aspectos hay que potenciar.

Sin embargo, la obtención de estos indicadores suele resultar costosa tanto en tiempo como en dinero, puesto que requiere de la recolección y posterior agregación de datos relevantes por parte de gran cantidad de entidades (escuelas, hospitales, oficinas de desempleo, etc.), o la realización de encuestas pormenorizadas a una parte significativa de la población. Además, en muchos países en vías de desarrollo resulta prácticamente imposible acceder a este tipo de información.

La creciente expansión de las redes sociales, junto con la generalización del uso de dispositivos móviles, que habitualmente disponen de conexión a internet, puede resultar una gran ventaja a la hora de abordar este problema. La traza de actividad que deja un usuario a lo largo del día puede dar una indicación sobre su nivel de vida o su ocupación. Por ejemplo, en [1] midieron la diversidad geográfica en la red de llamadas telefónicas en Reino Unido en agosto de 2005 y constataron que estaba fuertemente relacionada con el índice de desarrollo económico de cada comunidad, y en [2] utilizan indicadores agregados provenientes de los teléfonos móviles (relacionados con patrones de movilidad, llamadas y mensajes) para predecir el nivel socioeconómico de la población.

Uno de los indicadores que con más frecuencia se intenta estimar es la tasa de desempleo. En [3] lo hacen analizando mensajes de Twitter y midiendo cuántos de ellos contienen términos relacionados con la pérdida o la búsqueda de empleo; por otra parte, en el reciente estudio descrito en [4] se utilizan datos provenientes de teléfonos móviles (volumen de llamadas, usuarios contactados, usuarios con los que se ha perdido el contacto y radio de movimiento, entre otros) para reforzar modelos predictivos de series temporales, con lo que se consigue estimar el paro con bastante exactitud.

En esta línea, los autores de [5] proponen una prueba de concepto para la estimación del desempleo a partir de medidas agregadas obtenidas de redes sociales. En su estudio utilizan datos extraídos de la red social Twitter en España durante un periodo de siete meses, y concluyen que la mayoría de las variables propuestas contienen información moderada sobre la tasa de desempleo en las distintas zonas del país. Este estudio se describe en detalle en la sección 2.3.

1.2 Objetivos

El objetivo de este trabajo es extender el estudio llevado a cabo en [5] sobre la predictibilidad del paro a partir variables extraídas exclusivamente de redes sociales. Usando una gran base de datos de tweets geolocalizados en España durante dos años, se persigue:

1) Diseñar y crear un sistema eficiente que automatice el proceso de limpieza de los datos y generación de variables a partir de los mismos. Esto es especialmente importante debido a la gran cantidad de datos y a la necesidad de prototipar y testear rápidamente los modelos.
2) Generar un grafo de movilidad en España a partir de los datos de Twitter y estudiar y aplicar técnicas de detección de comunidades en grafos para identificar áreas cohesionadas dentro del territorio.
3) Estudiar el grado de información acerca del paro que contienen las variables calculadas, y cómo varía éste en función del tiempo y de las decisiones tomadas acerca de la forma de generar los indicadores.
4) Estudiar distintas posibilidades de predicción para el nivel de desempleo en las distintas zonas geográficas a lo largo del tiempo, y construir y analizar diferentes modelos predictivos que traten de estimar el paro utilizando las variables extraídas de Twitter.

1.3 Estructura de esta memoria

El documento se ha dividido en seis secciones, en cada una de las cuales se presenta una parte del trabajo realizado.

En la primera sección, Introducción, se explican la motivación del trabajo y los objetivos que se persiguen.

En la segunda sección, Estado del arte, se presentan los principales resultados teóricos que se utilizarán en este estudio, adquiridos durante la etapa universitaria, o bien a través de un proceso adicional de documentación.

Seguidamente se suceden cuatro secciones que resumen el trabajo realizado en este proyecto, dividiéndolo en varios aspectos a considerar.

En la tercera sección, Herramientas y tecnologías, se describen brevemente las principales herramientas utilizadas en el trabajo que resultaban novedosas.

En la cuarta sección, Diseño y desarrollo, se analizan las necesidades del sistema y se describe la implementación realizada.

En la quinta sección, Pruebas y resultados, se muestran los principales análisis realizados en el estudio y se describen los resultados de los distintos modelos predictivos utilizados.

Por último, en la sexta sección, Conclusiones y trabajo futuro, se exponen las conclusiones extraídas de este trabajo y las posibles extensiones que podría tener.

2 Estado del arte

En esta sección se explican los conceptos y resultados en los que se basa este trabajo. Primero se describen las técnicas de análisis de redes sociales y teoría de grafos que se han utilizado, después se introducen unas nociones de modelos predictivos y series temporales, y por último se resume el trabajo realizado en [5], que sirve de punto de partida para este estudio.

2.1 Análisis de redes sociales

Existen una gran cantidad de sistemas basados en conexiones: desde relaciones humanas hasta procesos biológicos o de transferencia de información. Las redes resultan extremadamente útiles para modelar estos sistemas, y su análisis como entidad teórica puede proporcionar conocimiento que ayude a gestionarlos. Para esto se utiliza la Teoría de Grafos, un campo de estudio iniciado por Euler en el siglo XVIII con el famoso problema de los puentes de Königsberg, y que actualmente recibe numerosas aportaciones debido en gran medida al auge de las redes sociales y a la mejora de las capacidades computacionales, que permiten realizar análisis más complejos y costosos.

2.1.1 Un ejemplo de red social: Twitter

Twitter se define como una red social online de microblogging. Los usuarios tienen la posibilidad de escribir pequeños mensajes, llamados **tweets**, de 140 caracteres como máximo, en los que suelen plasmar sus opiniones e inquietudes. Existe la posibilidad de referenciar a otro usuario en nuestros mensajes, escribiendo su alias precedido por un símbolo @; esto se conoce como **mención**. Es posible también resaltar palabras clave poniendo un símbolo # delante; esto se denomina **hashtag** y permite agrupar tweets que hablen de temas parecidos. Finalmente, Twitter ofrece la posibilidad de **geolocalizar** tus tweets, esto es, indicar en el mensaje la localización espacial en la que se encuentra el usuario en el momento de escribirlo. Los usuarios pueden interaccionar con mensajes ajenos respondiendo, retwitteando (compartiendo ese mensaje con sus seguidores), o marcándolos como favorito, quedando entonces guardados en una lista accesible desde su perfil. Pero lo que hace especial a Twitter es el hecho de que la gran mayoría de los perfiles están abiertos y pueden ser leídos por todo el mundo, y que dispone de una API de Streaming de muy fácil uso que permite extraer información pormenorizada sobre los mensajes y los perfiles de usuarios. También existe una API REST que se usa en el desarrollo del lado cliente porque permite además interactuar como usuario (enviar mensajes, seguir a un usuario, etc.), pero que no resulta útil para extraer gran cantidad de datos para su análisis puesto que las búsquedas están optimizadas para mostrar únicamente una porción de los tweets (los más recientes o los más relevantes). Ambas APIs proporcionan la información (tweets, usuarios, lugares, etc.) en formato JSON.

2.1.2 Detección de comunidades

Uno de los problemas más abordados en el análisis de redes, no necesariamente sociales, es la detección de comunidades. La mayoría de las redes objeto de análisis muestran cierto agrupamiento de sus nodos, grupos que por lo general vienen determinados por la disposición de las aristas entre los vértices. Esto no ocurre en las redes en las que el número de aristas es muy superior al número de nodos, puesto que en ese caso la cohesión es muy grande entre todos los nodos y no existen particiones relevantes.

La identificación de estos grupos o comunidades puede resultar de gran utilidad para entender el funcionamiento de la red. Un ejemplo muy común para ilustrar este hecho es el del club de kárate de Zachary [6]: el presidente y el instructor de este club tenían ciertas diferencias que terminaron con el instructor creando un nuevo club por su cuenta, y llevándose consigo una parte de sus antiguos alumnos. En 1977, Zachary construyó un grafo que reflejaba las amistades entre los distintos miembros del club (incluidos el presidente y el instructor) y mediante un algoritmo de detección de comunidades fue capaz de inferir con un 97% de acierto la separación ocurrida en el club.

Desafortunadamente, no hay una definición formal de comunidad universalmente aceptada, pero sí parece claro que los nodos que formen una comunidad han de tener una **densidad de conexiones** entre ellos superior a la que muestren con nodos que se encuentren fuera de ella. Las comunidades pueden tener tamaños y densidades distintas, lo cual, junto con el hecho de que normalmente se desconoce a priori el número de grupos existentes en el grafo, complica enormemente la tarea de detección de comunidades. No obstante, esta área está siendo estudiada por una parte notable de la comunidad científica y se han propuesto multitud de algoritmos, parte de los cuales funcionan razonablemente bien en un tiempo de ejecución asumible.

Dentro del problema de detección de comunidades existen además algunas variantes. Los grafos pueden ser no dirigidos o dirigidos, y la información adicional que proporcione la dirección de las aristas puede ser muy relevante. Por tanto, las técnicas utilizadas para detección de comunidades en grafos no dirigidos difieren notablemente de las técnicas que se emplean en el caso dirigido. También puede ser que las aristas del grafo tengan pesos asignados, pero afortunadamente la mayoría de algoritmos están adaptados para este tipo de grafos. Por otra parte, es posible que en lugar de necesitar una partición al uso de los vértices de la red, donde cada nodo pertenezca a un único cluster, resulte deseable obtener un recubrimiento en el que cada nodo pueda pertenecer a varios grupos, existiendo por tanto solapamientos entre las comunidades detectadas. Esto ocurre en muchas redes de la vida real; por ejemplo, una misma persona puede pertenecer a varios grupos de amigos, o un científico puede desarrollar su labor investigadora en varias áreas. Puede ocurrir también que los nodos del grafo sean de varias clases, existiendo aristas únicamente entre nodos de distinta clase; por ejemplo, un grafo que modele el comercio en un barrio puede tener dos tipos de nodos: las tiendas y los vecinos que compran en ellas, y las aristas relacionarán a las personas con los comercios a los que acuden. Para este tipo de grafos también son necesarios algoritmos expresamente diseñados para resolver este problema. Por las necesidades de este trabajo, nos

centraremos en los algoritmos que generan una partición de la red en comunidades disjuntas y diseñados para grafos con un único tipo de nodo.

Definiciones de comunidad

S. Fortunato realiza en [7] un buen resumen del trabajo llevado a cabo en torno al problema de detección de comunidades. Para empezar, agrupa las distintas definiciones de comunidad en tres clases: locales, globales, y basadas en similitudes entre los vértices; esto resulta de gran ayuda para profundizar en la intuición de comunidad y conocer algunas de las técnicas más extendidas.

- **Definiciones locales**. Estas definiciones se basan en la evaluación de la cohesión de la comunidad independientemente del resto del grafo, aunque quizás teniendo en cuenta los vecinos más próximos. Dentro de este grupo identifica varios criterios. Nótese que en las definiciones el subgrafo en cuestión ha de ser maximal, puesto que si existe otro subgrafo con la propiedad que estamos presentando y que contenga al primero, la comunidad será en realidad ese subgrafo de mayor tamaño.
  - La definición más estricta es considerar como comunidad únicamente los grupos de nodos con aristas existentes entre todos ellos. Este tipo de subgrafo se denomina *clique* en Teoría de Grafos. Se puede relajar un poco considerando *n-cliques*, que son subgrafos en los que la distancia entre dos vértices cualesquiera es como máximo n.
  - Un subgrafo cohesionado podrá no ser una buena comunidad si el número de enlaces al exterior es también muy alto; deberá tener además pocas aristas que la conecten con el resto del grafo. Por tanto, se suelen **comparar las cohesiones interna y externa del subgrafo**. Una opción es comprobar la robustez del subgrafo frente a eliminación aleatoria de enlaces comparando la conectividad (mínimo número de aristas que hay que eliminar para que dos nodos queden desconectados) entre vértices internos y mixtos. Otra posibilidad es medir las densidades *intra-cluster* e *inter-cluster*, que se definen de la siguiente forma, donde $\mathcal{C}$ es el subgrafo, n_c es su número de vértices y n es el número de vértices del grafo completo:

$$\delta_{int}(\mathcal{C}) = \frac{\#\{\text{aristas internas de } \mathcal{C}\}}{\mathrm{n_c}\,(\mathrm{n_c}-1)/2} \qquad \delta_{ext}(\mathcal{C}) = \frac{\#\{\text{aristas que salen de } \mathcal{C}\}}{\mathrm{n_c}\,(\mathrm{n}-\mathrm{n_c})}$$

- **Definiciones globales**. En este caso, las comunidades se definen de forma conjunta. Las definiciones de este estilo más frecuentes se basan en la idea de que un grafo que tenga estructura de comunidades ha de ser muy diferente a un grafo aleatorio, puesto que en estos últimos todos los pares de nodos tienen igual probabilidad de estar conectados entre sí y por tanto la cohesión es muy homogénea. Se crea un **modelo nulo** con las mismas características estructurales del grafo en cuestión, pero generado de forma aleatoria. El modelo más popular es el propuesto por Newman y Girvan en [8], que se construye reconectando al azar las aristas del grafo original, pero con la condición de que el grado esperado de cada nodo en el modelo nulo coincida con el grado del nodo en el grafo original.

- **Definiciones basadas en similitudes entre los vértices.** La idea básica es agrupar vértices con características parecidas para construir las comunidades.
  - Si los vértices del grafo pueden colocarse en el espacio Euclídeo, la medida de similitud podrá ser cualquier norma (L_1, L_2, L_∞,…).
  - En los casos en los que esto no sea posible será necesario utilizar otra medida de similitud que se base en las relaciones de adyacencia entre los vértices. Una posibilidad es definir una distancia basándose en las similitudes entre vecinos:

$$d_{ij} = \sqrt{\sum_{k \neq i,j} (A_{ik} - A_{jk})^2} \quad (1)$$

    donde A es la matriz de adyacencia del grafo.
  - Otras técnicas para medir la similitud se valen de los caminos aleatorios en grafos. Un **camino aleatorio** sobre un grafo es una trayectoria que se construye partiendo de un vértice, desplazándose a cualquiera de sus vecinos con probabilidad proporcional al peso de la arista que los une, y continuando este proceso iterativamente. Este tipo de métricas vienen determinadas por la forma en que los caminos aleatorios conectan los dos nodos cuya similitud se pretende medir; por ejemplo, el número medio de pasos que ha de dar un caminante aleatorio que parta de uno de los vértices para llegar al otro vértice y volver al primero, o la probabilidad de visitar el segundo vértice en menos de un número de pasos determinado.

Modularidad: medida de calidad

Como técnica para definir comunidades de forma global mencionábamos el modelo nulo propuesto por Newman y Girvan en [8]. Utilizan dicho modelo para crear una métrica, llamada **modularidad**, con la que poder determinar la calidad de una partición concreta del grafo. Ésta es, con diferencia, la medida de calidad más extendida. En [9] realizan una construcción muy clara de la modularidad, que se desarrolla a continuación.

Sea A la matriz de adyacencia del grafo, y c_v la comunidad a la que pertenece el vértice v. La fracción de aristas que conectan vértices de la misma comunidad es

$$\frac{\sum_{vw} A_{vw}\, \delta(c_v, c_w)}{\sum_{vw} A_{vw}} = \frac{1}{2m} \sum_{vw} A_{vw}\, \delta(c_v, c_w)$$

donde $\delta(i,j)$ es 1 si $i = j$ y 0 en caso contrario, y $m = \frac{1}{2}\sum_{vw} A_{vw}$ es el número de aristas en el grafo. Esta fracción (un valor entre 0 y 1) deberá ser alta en las particiones buenas, pero no es una condición suficiente para determinar la calidad de una partición, puesto que toma su valor más alto en el caso trivial de un grafo con una única comunidad.

Para obtener una métrica útil se resta el valor que tomaría esta expresión en el caso de que la red fuese el modelo nulo definido en [8] y explicado anteriormente. Llamemos k_v al número de aristas que conectan el nodo v con otros nodos: $k_v = \sum_w A_{vw}$. La probabilidad de que exista una arista entre los vértices v y w si las conexiones se realizan

aleatoriamente pero respetando los grados de los vértices es $k_v k_w/2m$. Por este motivo, la modularidad se define de la siguiente forma:

$$Q = \frac{1}{2m}\sum_{vw}\left[A_{vw} - \frac{k_v k_w}{2m}\right]\,\delta(c_v, c_w) \tag{2}$$

Así, si la fracción de aristas que quedan dentro de las comunidades es similar a la que se esperaría en el modelo nulo, la modularidad estará cercana a 0. Si, por el contrario, el número de aristas *intra-comunidad* es superior a lo esperado en el grafo aleatorizado, esta medida será positiva.

Se puede simplificar esta expresión definiendo

$$e_{ij} = \frac{1}{2m}\sum_{vw} A_{vw}\,\delta(c_v, i)\,\delta(c_w, j)$$

que representa la fracción de aristas que unen vértices pertenecientes a la comunidad i con vértices de la comunidad j. Esto define una matriz e, siendo $traza(e) = \sum_i e_{ii}$ la fracción de aristas en la red que conectan vértices de la misma comunidad. Además, es posible reescribir $\delta(c_v, c_w) = \sum_i \delta(c_v, i)\delta(c_w, i)$. Por tanto, podemos desarrollar (2):

$$\begin{aligned} Q &= \frac{1}{2m}\sum_{vw}\left[A_{vw} - \frac{k_v k_w}{2m}\right]\sum_i \delta(c_v, i)\delta(c_w, i) \\ &= \sum_i\left[\frac{1}{2m}\sum_{vw} A_{vw}\delta(c_v, i)\delta(c_w, i) - \frac{1}{2m}\sum_{vw}\frac{k_v k_w}{2m}\delta(c_v, i)\delta(c_w, i)\right] \\ &= \sum_i\left[e_{ii} - \frac{1}{2m}\sum_{vw}\frac{k_v k_w}{2m}\delta(c_v, i)\delta(c_w, i)\right] \end{aligned} \tag{3}$$

Falta ahora simplificar el segundo término del sumatorio. Notemos que

$$\frac{1}{2m}\sum_{vw}\frac{k_v k_w}{2m}\delta(c_v, i)\delta(c_w, i) = \left(\frac{1}{2m}\sum_v k_v\delta(c_v, i)\right)\left(\frac{1}{2m}\sum_w k_w\delta(c_w, i)\right)$$

y definamos

$$a_i = \frac{1}{2m}\sum_v k_v\delta(c_v, i)$$

que representa la fracción de aristas que salen de la comunidad i.

Entonces ya podemos obtener la expresión simplificada de la modularidad a partir de (3):

$$Q = \sum_i\left[e_{ii} - \left(\frac{1}{2m}\sum_v k_v\delta(c_v, i)\right)^2\right] = \sum_i (e_{ii} - {a_i}^2) \tag{4}$$

Cabe mencionar que en [8], el artículo original, los autores definen la modularidad de forma más directa usando desde el inicio la matriz e y el vector a. A modo orientativo

indican que las redes con una fuerte estructura en comunidades suelen tener una modularidad de entre 0.3 y 0.7, y que valores superiores a 0.7 son poco frecuentes.

La métrica Q es adecuada para grafos sin pesos en las aristas y no dirigidos, pero se puede adaptar fácilmente para otros grafos más complejos. Si el grafo es **ponderado**, la probabilidad de existencia de aristas para el modelo nulo en (2) deberá ser $s_v s_w / 2w$, donde s_i es la suma de los pesos de las aristas adyacentes al vértice i, y w es la suma de los pesos de todas las aristas del grafo. De la misma manera, la matriz de adyacencia A debe remplazarse por W, matriz simétrica que en la posición vw contiene el peso de la arista que une los vértices v y w. Por tanto, la modularidad será

$$Q_w = \frac{1}{2w} \sum_{vw} \left[W_{vw} - \frac{s_v \; s_w}{2w} \right] \delta(c_v, c_w) \tag{5}$$

Si el grafo es **dirigido**, la probabilidad anterior depende del grado exterior del vértice v (número de aristas que salen de él) y del grado interior del vértice w (número de aristas que entran en él); como ahora la suma de grados interiores/exteriores es m, la modularidad se expresa de la siguiente forma:

$$Q_d = \frac{1}{m} \sum_{vw} \left[A_{vw} - \frac{k_v^{out} \; k_w^{in}}{m} \right] \delta(c_v, c_w) \tag{6}$$

Finalmente, si el grafo es tanto ponderado como dirigido, la modularidad será

$$Q_{wd} = \frac{1}{w} \sum_{vw} \left[W_{vw} - \frac{s_v^{out} \; s_w^{in}}{w} \right] \delta(c_v, c_w) \tag{7}$$

Métodos de detección de comunidades

Mencionaremos ahora, a modo de resumen del esquema desarrollado en [7], algunos de las metodologías que se emplean en el problema de detección de comunidades. Los algoritmos que se han utilizado en este trabajo, que se detallan más adelante, utilizan una o varias de las técnicas que se describen a continuación.

- **Clustering jerárquico**. Este tipo de algoritmos se basa en la definición de una medida de similitud entre vértices. Existen dos categorías de algoritmos:
    - Algoritmos **aglomerativos**, en los que los clusters se van uniendo iterativamente si la medida de similitud entre ellos es suficientemente alta. Es necesario por tanto extender la medida de similitud para poder aplicarla a los clusters. El proceso finaliza cuando se forma un único grupo que englobe todos los nodos.
    - Algoritmos **divisivos**, en los que los clusters se van dividiendo mediante la eliminación de aristas que unan vértices con similitud suficientemente baja. El proceso continúa hasta que existen tantos clusters como nodos en el grafo.

Los resultados de estos dos procedimientos se pueden ilustrar mediante **dendrogramas**, gráficos que representan las aglomeraciones o divisiones realizadas entre los nodos en cada iteración.

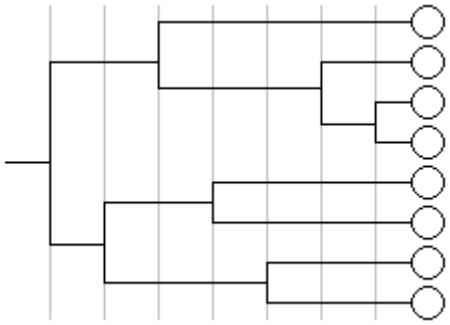

Figura 1 - Ejemplo de dendrograma

El clustering jerárquico en sí no proporciona un método para saber en qué iteración parar el proceso y por tanto es fundamental elegir una buena condición de parada que detecte en la medida de lo posible la partición óptima.

- **Métodos divisivos**. Estos algoritmos detectan las aristas que conectan vértices pertenecientes a distintas comunidades y las eliminan iterativamente. La diferencia principal con los métodos jerárquicos descritos en el punto anterior radica en que las métricas en este caso están definidas sobre aristas en lugar de pares de vértices. Sin embargo, los resultados también se representan mediante dendrogramas.

  Las aristas a eliminar son determinadas de acuerdo con los valores de una **medida de centralidad** de aristas; la más popular, usada en [8] para probar este tipo de algoritmos, es la intermediación (más conocida por el término inglés, *betweennes*). La definición más simple de intermediación de una arista (*edge betweennes*) es el número de geodésicas (caminos más cortos) que pasan por dicha arista; pero existen otras formas algo diferentes de definirla, como por ejemplo, la fracción de caminos aleatorios entre pares de vértices que pasan por esa arista. Intuitivamente parece claro que una arista con un alto grado de betweennes está actuando como "puente conector" entre varias secciones del grafo.

- **Optimización de la modularidad**. Este método es con diferencia, el más popular entre los algoritmos de detección de comunidades. Se centra en intentar encontrar la partición que maximice la modularidad, puesto que previsiblemente será la mejor partición, o al menos una de las mejores. Sin embargo, dado el enorme número de formas posibles de dividir un grafo, es imposible llevar a cabo una optimización exhaustiva de esta métrica. Por este motivo, los algoritmos propuestos se centran en encontrar una buena aproximación en un tiempo razonable, generalmente mediante **técnicas voraces**, que consisten en realizar iterativamente una serie de operaciones sobre el grafo, de forma que la operación escogida en cada paso sea la que suponga un mayor incremento de la modularidad, y siendo el resultado final una partición que se supone óptima.

- **Caminos aleatorios**. En una red con una fuerte estructura de comunidades, es de esperar que un caminante aleatorio pase una gran cantidad de tiempo dentro de una comunidad debido a la alta densidad de aristas dentro de ella; por tanto, los caminos aleatorios también pueden resultar útiles para detectar comunidades en un grafo. Generalmente se utilizan, como hemos mencionado con anterioridad, para definir medidas de distancia entre vértices, cuyos valores son usados posteriormente para obtener las comunidades. Una vez calculadas las distancias, las comunidades suelen extraerse usando alguno de los otros métodos descritos, como algoritmos jerárquicos u optimización de la modularidad.

Algoritmos de detección de comunidades

Como explicaremos más adelante, en este trabajo era necesario detectar la estructura de comunidades de un grafo de movilidad. Para ello se han probado seis conocidos algoritmos de detección de comunidades con tiempo de ejecución casi lineal para grafos dispersos, es decir, con un número moderado de aristas, y se han analizado sus resultados para decidir cuál era el más conveniente. Los algoritmos son los siguientes:

- **Fast Greedy** [9]. Es una mejora computacional de un algoritmo voraz de optimización de la modularidad propuesto por Newman en 2004. Este algoritmo parte de una situación en la que cada vértice es una comunidad, e iterativamente une las comunidades cuya fusión produce un mayor aumento de Q, hasta que acaba con una única comunidad formada por todos los nodos. Se trata por tanto de un algoritmo jerárquico aglomerativo que utiliza la optimización de la modularidad para decidir qué comunidades se unen en cada caso. Los autores asumen que la red, con n nodos y m aristas, va a ser dispersa ($m \sim n$) y jerárquica (dendrograma balanceado); esto les permite utilizar estructuras de datos más eficientes, lo que da lugar a una mejora considerable en el tiempo de ejecución.

- **Infomap** [10]. Se trata de un algoritmo para grafos ponderados y dirigidos; estos grafos representan interacciones entre las unidades de un sistema y, por tanto, para extraer la estructura en comunidades es necesario entender el flujo de información de la red. Los autores utilizan caminos aleatorios de longitudes grandes como representación de dichos flujos y, asignando una etiqueta distinta a cada nodo, desarrollan un código que describe estos caminos. Identificar las comunidades se traduce entonces en un problema de optimización de la compresión de la información: es deseable encontrar la mejor forma de comprimir el mensaje, de forma que la pérdida de información sea la menor posible; cada segmento comprimido, que estará compuesto por las etiquetas de varios nodos, definirá una comunidad distinta en el grafo.

- **Label Propagation** [11]. Este algoritmo utiliza la estructura del grafo para ir guiando su progreso, en lugar de tratar de optimizar alguna métrica como suele ser la modularidad. Los autores parten de la idea de que cada nodo de la red "elige" pertenecer a la comunidad a la que pertenecen la mayoría de sus vecinos; por tanto, su método consiste en asignar a cada nodo una etiqueta y dejar que se vaya

propagando a través del grafo siguiendo esa premisa. De esta forma, los grupos densamente conectados acaban llegando a un consenso en la etiqueta tras cierto número de iteraciones. El orden en el que cada uno de los nodos del grafo se va actualizando en cada iteración se elige de forma aleatoria, y el algoritmo termina cuando, para cada nodo, la etiqueta que éste lleva es mayoritaria entre sus vecinos.

- **Multilevel** [12]. Este algoritmo utiliza el método de optimización de la modularidad, en donde en cada iteración hay dos fases. Primero, se estudia secuencialmente cada nodo y se calcula la ganancia de modularidad que conllevaría eliminar dicho nodo de la comunidad a la que pertenece y añadirlo a la comunidad de uno de sus vecinos; el nodo se cambia a la comunidad vecina con la que se obtenga mayor ganancia, siempre y cuando ésta sea positiva. Esta primera fase termina cuando no existen más movimientos que puedan mejor la modularidad, es decir, cuando se haya llegado a un máximo local. La segunda parte consiste sencillamente en construir una nueva red en la que los nodos sean ahora las comunidades generadas en el paso anterior. Entonces se realiza una nueva iteración (compuesta de nuevo por las dos fases descritas), pero esta vez partiendo de la nueva red creada. Estas pasadas se repiten hasta que dejan de producirse cambios, momento en el que se habrá llegado a un máximo en la modularidad.

- **Walktrap** [13]. El enfoque de este algoritmo se basa en que los caminantes aleatorios tienden a quedarse "atrapados" dentro de las zonas con conexiones muy densas, puesto que en ese caso la probabilidad de moverse a un nodo que esté dentro de la comunidad es mayor que la de atravesar una arista que esté aislada. Con esto definen una distancia entre grafos que después utilizan para llevar a cabo un procedimiento jerárquico aglomerativo como los descritos con anterioridad. La distancia en cuestión es una adaptación de (1) que utiliza la matriz de transiciones P, donde $P_{ij} = A_{ij}/s_i$, es decir, la probabilidad de desplazarse del nodo i al nodo j. P^t es por tanto una matriz que define la probabilidad de llegar de un nodo a otro mediante un camino aleatorio de longitud t. La métrica de similitud entre vértices resulta por tanto de la siguiente forma:

$$r_{ij} = \sqrt{\sum_{k \neq i,j} \frac{(P_{ik}^t - P_{jk}^t)^2}{s_k}}$$

Es necesario adaptar esta métrica para medir distancias entre comunidades y poder así decidir qué par fusionar. Para ello se define la probabilidad P_{Cj}^t de ir desde la comunidad C hasta el nodo j en t pasos y se generaliza la distancia de la siguiente manera:

$$r_{C_1 C_2} = \sqrt{\sum_{k=1}^{n} \frac{(P_{C_1 k}^t - P_{C_2 k}^t)^2}{s_k}}$$

Comparación de comunidades

Para terminar con la sección referente a detección de comunidades, es necesario mencionar la existencia de métricas que indican el grado de similitud entre dos particiones. Esto puede resultar útil por ejemplo para comprobar lo bien que se comporta un determinado algoritmo de detección de comunidades en el caso de que se conozca a priori la partición óptima de un grafo de ejemplo.

La métrica que se ha utilizado en este trabajo recibe el nombre de ***Normalised Mutual Information*** (NMI) [14]. Los autores enfocan este problema como una forma de medir el nivel de acierto de un algoritmo de detección de comunidades, considerando que existe una partición correcta A del grafo, y otra partición B que se desea evaluar. La definición de esta métrica se basa en una matriz de confusión N en la que las filas corresponden a las comunidades reales y las columnas a las comunidades determinadas por el algoritmo, y donde la celda N_{ij} representa el número de nodos que pertenecen a la comunidad real i y han sido asignados a la comunidad definida j. Definen entonces su métrica como

$$I(A,B) = \frac{-2\sum_{i=1}^{k_A}\sum_{j=1}^{k_B} N_{ij}\log\left(\frac{n\cdot N_{ij}}{a_i b_j}\right)}{\sum_{i=1}^{k_A} a_i \log\left(\frac{a_i}{n}\right) + \sum_{j=1}^{k_B} b_j \log\left(\frac{b_j}{n}\right)} \tag{8}$$

donde k_A y k_B indican el número de comunidades de las particiones A y B, $n = \sum_{ij} N_{ij}$ es el número total de nodos del grafo, $a_i = \sum_j N_{ij}$ es el número de nodos en cada comunidad i de la partición A, y $b_j = \sum_i N_{ij}$ es el número de nodos en cada comunidad j de la partición B.

Para entender la idea detrás de esta definición, notemos que guarda una gran similitud con la **información mutua** de dos variables aleatorias X, Y en Teoría de la Información, que mide cuánto reduce la incertidumbre sobre una variable el hecho de conocer información sobre la otra, esto es, la información que las dos variables comparten. Se define:

$$I(X,Y) = \sum_{x\in X}\sum_{y\in Y} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

En (8),

$$p(i,j) = {}^{N_{ij}}\!/_{n}$$

$$p(i) = {}^{a_i}\!/_{n}$$

$$p(j) = {}^{b_i}\!/_{n}$$

$$\log\left(\frac{p(i,j)}{p(i)p(j)}\right) = \log\left(\frac{{}^{N_{ij}}\!/_{n}}{{}^{a_i}\!/_{n}\,{}^{b_j}\!/_{n}}\right) = \log\left(\frac{n\cdot N_{ij}}{a_i b_j}\right),$$

y esto es lo que aparece en el numerador. El resto de factores, que se simplifican ligeramente al dividir por la n de $p(i,j)$, simplemente normalizan para que esta medida no dependa de los tamaños de las particiones.

2.2 Modelos predictivos

El problema de predicción consiste en inferir cierta información sobre la variable aleatoria Y a partir de un vector aleatorio formado por p variables aleatorias. La variable Y recibe el nombre de **variable respuesta** o variable dependiente, y las p variables del vector suelen ser denominadas sencillamente variables del modelo.

En la práctica se dispone de n observaciones del vector aleatorio X, es decir, de una matriz $n\,x\,p$ con un valor de cada una de las variables para cada observación, y se desea determinar el valor de Y correspondiente para cada observación. Los modelos predictivos usan un conjunto de ejemplos (observaciones de X con su respectivo valor de Y) para **entrenar**, esto es, determinar los parámetros que mejor ajustan la predicción al valor real de Y, y posteriormente estos parámetros se pueden usar para **predecir** el valor de la variable respuesta para nuevas observaciones de X, con cierto grado de error.

Se distinguen dos tipos de problemas en función de la naturaleza de los valores Y. Pueden consistir en una serie de k etiquetas prefijadas, y en ese caso será un **problema de clasificación**. Por ejemplo, se puede desear predecir la categoría a la que pertenece un correo electrónico (trabajo, ocio, spam) a partir de ciertas variables como pueden ser la hora de recepción, la presencia de determinadas palabras clave, etc. Para poder funcionar correctamente, el modelo deberá haber sido entrenado con ejemplos de las k etiquetas que luego va a tener que considerar como opciones de salida. Esto suele simplificarse en k problemas de clasificación binaria, con Y pudiendo pertenecer a dos clases, 0 ó 1, en los que se determina la probabilidad de pertenecer a la clase 1: $P(Y = 1 \mid X)$. Algunos ejemplos de algoritmos de clasificación son el de Naïve-Bayes, k-NN (*Nearest Neighbours*), árboles de decisión, *Support Vector Machines* (SVM) o perceptrones, implementados mediante redes neuronales.

Otra opción es que Y pueda tomar cualquier valor numérico en una escala continua, que el modelo deberá ser capaz de inferir aunque no haya entrenado con ningún ejemplo con exactamente el mismo valor. Este caso se denomina **problema de regresión**, consistente en calcular $\mathbb{E}(Y \mid X) = f(X, \beta) + \varepsilon$ como función de los valores de X y de una serie de coeficientes β que se determinan mediante el entrenamiento; la función depende del algoritmo de regresión y siempre se comete cierto error ε como consecuencia de la aleatoriedad de Y. Este es el tipo de problema que se va a considerar en este estudio, puesto que el nivel de desempleo puede tomar valores en una escala continua. Sin embargo, los valores del paro están comprendidos entre 0 y 100, mientras que los modelos regresores no tienen ninguna limitación; esto puede suponer un problema a la hora de predecir, y la solución por la que se suele optar es aplicar alguna función asintótica a la salida del modelo para ubicar los valores dentro del intervalo deseado. Algunos ejemplos de algoritmos de regresión son: regresión lineal (multivariante), *Support Vector Regression* (SVR), *Ridge Regression*, modelos implementados en redes neuronales, o modelos autorregresivos (AR, ARMA, ARIMA, de los que hablaremos en la siguiente sección2.2.1).

2.2.1 Modelos de regresión en series temporales

La predicción de series temporales es un problema que hay que abordar de forma algo diferente, puesto que en una serie temporal es muy probable que los valores en un instante estén en cierta medida determinados por los valores que ha tomado la serie con anterioridad.

Las series temporales pueden separarse en tres componentes: la **componente *trend***, que representa la tendencia de la serie, la **componente *seasonal***, que corresponde con la parte estacionaria, y la **componente *random***, que recoge las variaciones en la serie no explicadas por las otras dos componentes. En la Figura 2 se muestra un ejemplo de la descomposición realizada sobre la serie temporal del paro medio en España de la Encuesta de Población Activa, que se publica trimestralmente.

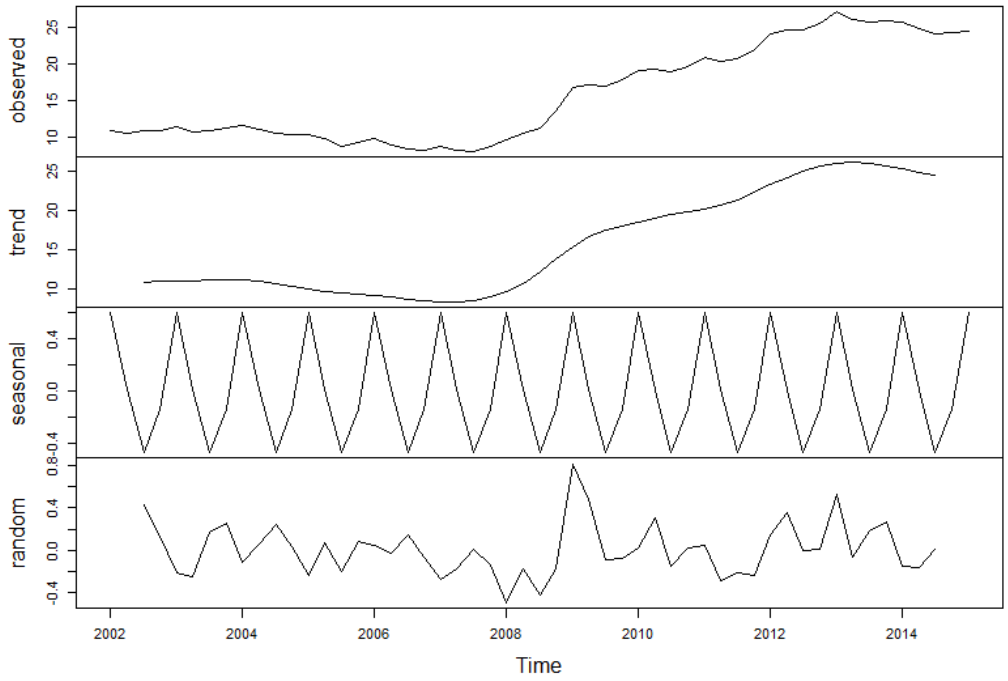

Figura 2 - Descomposición de la serie temporal del desempleo medio en España

Una opción para realizar predicciones sobre una serie temporal de este tipo es eliminar la componente estacional e intentar estimar las otras dos. Generalmente las variables que estimen mejor cada una de las componentes no serán las mismas, puesto que en la primera los cambios se suceden de forma muy suave, mientras que la segunda probablemente se ajuste mejor a variables con comportamientos más espontáneos. Si la predicción es a corto plazo, podremos considerar constante la componente *trend* y predecir únicamente la componente *random*.

Otra posibilidad es utilizar un **modelo autorregresivo (AR)**, en el que la variable respuesta Y depende exclusivamente de sus valores anteriores. El modelo autorregresivo de orden p, $AR(p)$, se define de la siguiente forma:

$$Y_t = \sum_{i=1}^{p} \phi_i Y_{t-i} + \varepsilon_t$$

Una forma de determinar el valor del parámetro p es analizando las **autocorrelaciones** de la serie, esto es, la correlación de la serie con versiones desplazadas de la misma. Por ejemplo, en el caso del paro de la EPA, la autocorrelación se va a cero al desplazar la serie cuatro periodos, tal y como se aprecia en la Figura 3; por tanto, lo mejor será usar un modelo con parámetro p inferior a 4: AR(1), AR(2) o AR(3).

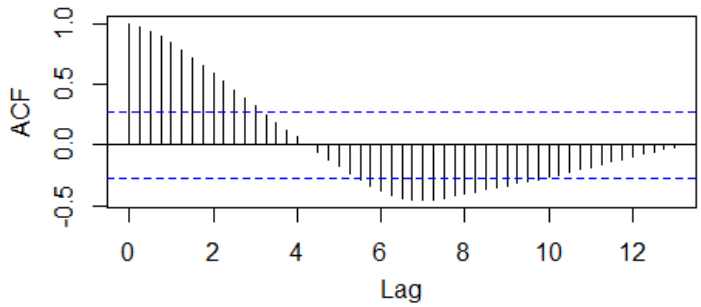

Figura 3 - Autocorrelaciones de la serie temporal del desempleo medio en España

El modelo se puede mejorar considerando además el error cometido en la predicción en periodos anteriores; esto se consigue añadiendo una parte de media móvil (MA) al modelo AR, lo que da lugar al modelo $ARMA(p, q)$, con expresión

$$Y_t = \sum_{i=1}^{p} \phi_i Y_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

Es posible además incluir variables exógenas en estos modelos. Si se sabe que la serie temporal Y a predecir está relacionada con otra serie temporal T, se puede usar un modelo $ARMAX(p, q, b)$ de expresión

$$Y_t = \sum_{i=1}^{p} \phi_i Y_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \sum_{i=1}^{b} \eta_i T_{t-i} + \varepsilon_t$$

Los modelos autorregresivos aceptan muchas más variantes que aumentan el nivel de complejidad y sofisticación, pero en este trabajo el límite se ha puesto en el modelo ARMA por los motivos que se explican en la sección 5.3.1.

2.2.2 Modelos de regresión generales

Además de los específicos para series temporales, en el trabajo se han utilizado otros dos modelos de regresión en el problema de predicción de la tasa de desempleo.

El modelo de **regresión lineal**, uno de los más sencillos, consiste en estimar la variable respuesta como una combinación lineal de las variables regresoras.

El modelo **random forest** consiste en entrenar un número n de árboles de decisión a partir de un subconjunto de las variables de tamaño fijo m pero escogido aleatoriamente y sacar como predicción el valor medio de los resultados de estos árboles. La predicción suele ser mejor cuanto mayor es n, pero m ha de ser mucho menor que el número total de variables.

2.3 Social Media Fingerprints of Unemployment

El estudio realizado en [5] se ha tomado como referencia para la realización de este trabajo y, por tanto, se describe con detalle en esta sección. Como hemos mencionado anteriormente, el objetivo de los autores era encontrar una serie de indicadores extraídos de redes sociales que sirviesen para aproximar el nivel de desempleo en una comunidad. La motivación no era tanto encontrar una forma de inferir el paro en España, donde ya se cuenta con métodos que proporcionan este dato con gran exactitud (por ejemplo, la Encuesta de Población Activa, EPA), sino más bien hallar una forma de determinar el nivel de desempleo a partir de unos datos fácilmente accesibles, que posteriormente pueda replicarse en otras zonas del planeta en las que estas estadísticas son más complicadas de obtener. Por ejemplo, actualmente estamos realizando un proyecto en colaboración con Pulse Lab Jakarta para tratar de adaptar este modelo al país de Indonesia, en el que el número de usuarios de redes sociales es relativamente alto pero los resultados de las encuestas de población activa son muy inexactos.

El estudio se centra en diseñar una serie de indicadores o variables que se puedan obtener íntegramente de las redes sociales, y que sirvan para estimar el nivel de desempleo. Para ello utilizan la red social Twitter, por el tipo de datos que proporciona y la facilidad con la que se obtienen. Existe sin embargo una desventaja, y es que Twitter es una de las redes menos fiables: una cantidad no despreciable de los usuarios son en realidad *bots* (cuentas ficticias controladas de forma automática con fines mayormente publicitarios), el porcentaje de usuarios que envía mensajes de forma regular no es muy elevado y la mayoría de los tweets no están geolocalizados. Aun así, como veremos, varias de las variables extraídas de esta red social proporcionan un nivel notable de información acerca de la variable respuesta, el nivel de desempleo.

Recopilan todos los tweets geolocalizados en España (excluida la zona de las Islas Canarias) en un periodo de tiempo comprendido entre el 29 de noviembre de 2012 y el 30 de junio de 2013. Esto suponen un total de 19.6 millones de tweets publicados por 0.57 millones de usuarios únicos. Cabe destacar que la correlación observada entre el número de tweets geolocalizados en cada municipio y la población correspondiente es muy elevada (0.951). Intuitivamente, esto parece indicar que los datos extraídos son representativos de la situación global en España.

La división municipal de la Península no parece muy adecuada para estudiar la actividad socioeconómica, puesto que en muchas ocasiones los lazos económicos y sociales traspasan estas divisiones por estar éstas determinadas por motivos más bien históricos y políticos. En lugar de utilizar otras divisiones territoriales predefinidas, A. Llorente *et. al.* optan por un procedimiento que les permite detectar comunidades cohesionadas en base a los datos de la red social. Se trata de construir una matriz de flujos, muy utilizada en teoría del transporte, que recoge el número de viajes diarios realizados entre cada pareja de municipios. Se considera que un usuario ha realizado un **viaje** entre el municipio *i* y el municipio *j* cuando ha tuiteado en los sitios *i* y *j* consecutivamente a lo largo del mismo día. De esta forma obtienen 1.9 millones de viajes realizados por 0.22 millones de

usuarios, con los que construyen la matriz de movilidad. Esta matriz se puede ver como la matriz de adyacencia asociada a un grafo dirigido, de manera que es posible definir las agrupaciones buscadas utilizando un algoritmo de detección de comunidades.

El algoritmo utilizado para este fin es Infomap, con el que detectan 340 comunidades distintas con una media de 21 municipios por comunidad. Estas comunidades tienen cuatro propiedades interesantes:

- Son cohesionadas desde un punto de vista geográfico.
- Son robustas frente a la eliminación aleatoria de viajes en la matriz.
- La medida de modularidad de la partición es alta (0.76).
- Las comunidades definidas tienen bastante solapamiento con las provincias administrativas (77% de NMI), y más aún con las comarcas (83% de NMI).

Por tanto, parece que las comunidades obtenidas mediante este procedimiento son una buena aproximación de las áreas económicas del país. A partir de esta constatación los autores se centran en estas comunidades a la hora de encontrar variables que aproximen el paro, que se agrega a nivel de estas comunidades socioeconómicas. Aun así, afirman que los resultados del estudio también se sostienen a nivel de municipio, comarca o provincia, aunque en menor medida.

Antes de pasar a describir las variables contempladas en su estudio es necesario definir el concepto de **hogar** de un usuario, que se utiliza en varias de ellas. Para tratar de identificar el municipio en el que vive cada usuario se cuentan el número de tweets que ha realizado en cada una de estas áreas y se le asigna aquel en el cuál haya tuiteado con más frecuencia, siempre y cuando la diferencia con el siguiente sea notable. Concretamente, se afirma que un municipio es el hogar de un usuario si este ha publicado más de 5 tweets geolocalizados en total y como mínimo el 40% de ellos están en ese municipio. Definen la población en Twitter π_i en el área i como el número de hogares localizados en dicha área, y encuentran una alta correlación (0.977) entre π_i y la población real P_i de cada área, lo que parece indicar que la distribución de lugares de residencia a lo largo de las zonas se corresponde de forma acertada con la realidad.

Las once variables que proponen para la estimación del paro se dividen en cuatro grupos, y se detallan a continuación. Estas variables obtienen un valor por cada una de las áreas.

- **Nivel de utilización de redes sociales**. Algunos estudios recientes muestran que existe una correlación a nivel de país entre la proporción de personas que utilizan Twitter y el PIB, por lo que una variable de este estilo puede resultar relevante.

    1) **Tasa de utilización de Twitter**: fracción τ_i de usuarios con hogar asignado en el área i entre la población registrada en dicho lugar.

- **Actividad diaria en redes sociales**. Es probable que áreas con diferentes niveles de desempleo muestren patrones de actividad distintos a lo largo del día, puesto que no todas las actividades se desarrollan a las mismas horas.

    2) **Fracción de tweets por la mañana**: proporción v_i^m de tweets publicados entre las 08h y las 10h sobre el total de tweets publicados en días de diario.

3) **Fracción de tweets por la tarde**: proporción v_i^t de tweets publicados entre las 15h y las 17h sobre el total de tweets publicados en días de diario.
4) **Fracción de tweets por la noche**: proporción v_i^n de tweets publicados entre las 00h y las 03h sobre el total de tweets publicados en días de diario.

- **Contenido de los mensajes en redes sociales**. Por una parte, cabe la posibilidad de que una persona que ha conseguido o perdido un empleo, o sencillamente que tiene un trabajo al que acude habitualmente, lo comunique a sus conocidos a través de mensajes en Twitter. Esta es la aproximación que utilizan en [3], como mencionamos anteriormente. Los autores prueban además una aproximación que trate de relacionar la forma de escribir con el nivel de desempleo: es posible que las zonas en las que se cometen más faltas de ortografía el nivel educativo sea más bajo y esto esté relacionado de alguna forma con la tasa de desempleo.

  5) **Fracción de tweets sobre *trabajo***: proporción μ_i^{tr} de tweets que mencionan "trabajo" o derivados sobre el total de tweets publicados.
  6) **Fracción de tweets sobre *paro***: proporción μ_i^{pa} de tweets que mencionan "paro", "despidos" o derivados sobre el total de tweets publicados.
  7) **Fracción de tweets sobre *economía***: proporción μ_i^{ec} de tweets que mencionan "economía" o derivados sobre el total de tweets publicados.
  8) **Fracción de tweets sobre *empleo***: proporción μ_i^{em} de tweets que mencionan "empleo" o derivados sobre el total de tweets publicados.

  9) **Fracción de personas que cometen faltas de ortografía**: proporción de personas ε_i que han publicado al menos un tweet con faltas de ortografía sobre el total de personas con hogar asignado en esa área. Para detectar faltas de ortografía comprueban si el texto de los tweets contiene alguna de las expresiones recogidas en una lista con 618 expresiones mal escritas en castellano que construyeron para este fin.

- Interacciones en redes sociales y diversidad del flujo geográfico. Como se estudió en [1], parece que la diversidad en las interacciones sociales suele venir de la mano de un mayor nivel socioeconómico. Esto es un hecho ampliamente aceptado en Teoría Social y puede resultar de utilidad para este problema (un patrón de comunicaciones diverso proporciona más oportunidades y por tanto una mayor empleabilidad). Para medir la diversidad utilizan, al igual que hacen en [1], la **entropía de información normalizada**

$$S_i = \frac{-\sum_j p_{ij} \cdot ln\, p_{ij}}{ln\, k_i}$$

  donde p_{ij} es la fracción de comunicaciones o viajes desde el área i que se producen hacia el área j, y k_i es el número de áreas con las que se relaciona el área i. Este estadístico, propuesto por Shannon en 1948, toma valores entre 0 y 1 y mide el grado de incertidumbre existente en una variable aleatoria, que en nuestro caso sería el área con la que se comunica el área i, o el área hacia la que se viaja desde el área i. Por tanto, intuitivamente parece claro que si se interacciona con pocas áreas la

incertidumbre de esta variable aleatoria será baja y por tanto la entropía será cercana a 0, y si el número de áreas con las que existe interacción es muy alta, la incertidumbre será muy alta y consecuentemente la entropía será cercana a 1. Por tanto, la entropía es un buen indicador del grado de diversidad del área.

10) **Entropía social**: entropía S_i de las menciones que se producen en tweets geolocalizados en el área i.
11) **Entropía de movilidad**: entropía $\tilde{S}_i$ de los viajes que se producen con localización de partida ubicada en el área i.

Una vez definidas las variables intentan determinar el grado de información que aportan. Para ello calculan la matriz de correlaciones entre ellas, con la que comprueban que existe una correlación moderada entre las variables pertenecientes al mismo grupo, y llevan a cabo un análisis de componentes principales, llegando a la conclusión de que cada uno de los grupos explica una fracción de la variabilidad del nivel de desempleo. Por tanto, escogen de cada grupo de variables aquellas que muestran mayor correlación con el desempleo: la tasa de utilización de Twitter τ_i, las variables de diversidad social (S_i) y de movilidad ($\tilde{S}_i$), la actividad por la mañana (υ_i^m), la fracción de usuarios que cometen faltas de ortografía (ε_i), y la fracción de tweets que hablan de empleo (μ_i^{em}).

Finalmente, utilizan estas seis variables para construir un modelo de regresión multivariante con el nivel de desempleo en cada zona como variable regresora. La variabilidad del nivel de paro que queda explicada por la regresión está en torno al 40% para el caso del paro total, y oscila entre el 50% y el 65% para el paro joven (de personas menores de 25 años), dependiendo del mes al que pertenezca el dato de desempleo.

Adicionalmente, un estudio sobre la importancia relativa de las variables en el modelo lineal revela que las variables más importantes son τ_i, ε_i, $\tilde{S}_i$ y υ_i^m, por este orden, mientras que las variables S_i y μ_i^{em} no son estadísticamente significativas para el modelo.

Por tanto, los autores encuentran que la tasa de utilización de Twitter, junto con la fracción de usuarios que cometen faltas de ortografía, la entropía de movilidad y la actividad por la mañana, todas ellas variables obtenidas de la red social Twitter, contienen información sobre el nivel de desempleo, especialmente el joven, en las distintas zonas de España.

3 Herramientas y tecnologías

3.1 Lenguaje de programación R

R ha sido el principal lenguaje de programación utilizado para el desarrollo, debido a la eficiencia con la que maneja conjuntos grandes de datos y a la facilidad para programar rutinas de procesamiento de datos (filtros, agregaciones, etc.) y para generar gráficas con mucha información. Sin embargo, R resulta extremadamente ineficiente en programación convencional (bucles, estructuras de datos personalizadas, y demás recursos que en lenguajes convencionales tipo C se utilizan constantemente). Por tanto, la forma de programar en R es muy diferente: rara vez se usan bucles y se aprovechan al máximo las librerías de manejo de datos en tablas, como *data.table*, *plyr* o *dplyr*.

A la hora de generar gráficos, la librería *ggplot* ha sido de gran utilidad. Utiliza un paradigma diferente al de las librerías convencionales, basado en la *gramática de los gráficos*, que divide un gráfico en varias componentes diferenciadas (capas, escalas, temas,…). Resulta algo complicado de utilizar al principio, pero su potencia es inmensa.

3.2 Manejo de información geoespacial

Dado que una cuestión a analizar sobre los tweets era su geolocalización, el manejo de información geoespacial (puntos, municipios) ha sido indispensable en este trabajo.

Twitter especifica las localizaciones usando el **formato GeoJSON**:

```
{ "type": <type>, "coordinates": <coordinates> }
```

donde `<type>` puede ser `"Point"`, `"Polygon"` o `"Multipolygon"`, entre otros, y `<coordinates>` es una lista con las coordenadas que definen a la forma geométrica; en el caso del punto, se trata sencillamente de la lista `[<lon>, <lat>]`, mientras que en los polígonos son una serie de listas anidadas que definen el polígono principal y los agujeros de su interior a través de secuencias de puntos: `[ [ [<lon1>, <lat1>], ..., [<lonN>, <latN>], [<lon1>, <lat1>] ], [ ... ], ... ]`. En los multipolígonos, las coordenadas se componen de una secuencia de coordenadas de polígono.

En R, el paquete *sp* contiene una serie de clases (*SpatialPoints*, *SpatialPolygons*,…) que modelan estas estructuras, a las que se puede agregar además una tabla con información extra (por ejemplo, el nombre del polígono si se trata de una división territorial). Adicionalmente, el paquete *raster* ofrece una serie de funciones para manejar estas clases (detectar solapamientos entre figuras, representarlas gráficamente, etc.).

3.3 Bases de datos NoSQL: Elasticsearch y MongoDB

El volumen de tweets a procesar y el hecho de que la API los proporcione en formato JSON permiten plantear la posibilidad de usar una **base de datos NoSQL basada en documentos**, en lugar de los habituales modelos relacionales. Este tipo de bases de datos

están pensadas para manejar grandes volúmenes de datos, mediante la distribución y la replicación automática de los mismos en las distintas máquinas disponibles, y además ofrecen mucha flexibilidad para añadir nueva información a los documentos y disponen de elaborados lenguajes para realizar consultas. Para este trabajo se han explorado las posibilidades de dos bases de datos de este tipo: Elasticsearch y MongoDB.

3.3.1 Elasticsearch

Elasticsearch [15] es un motor de búsqueda y análisis de datos en tiempo real que permite extraer información a una gran velocidad a través de una API REST. Los datos JSON se almacenan en documentos, que se agrupan en tipos, y estos a su vez en índices. Los documentos pertenecientes al mismo índice tienen campos idénticos o similares, que se pueden definir de antemano usando la API *mapping*. Por defecto, Elasticsearch indexa todos los datos que guarda para poder realizar búsquedas rápidamente; utiliza un índice invertido para desglosar los textos por palabras, lo que agiliza las búsquedas de contenido.

Las búsquedas y operaciones sobre los datos se realizan a través de la API *search*. Elasticsearch utiliza un lenguaje de consulta basado en JSON, **Query DSL**, que permite realizar dos tipos de búsquedas: ***queries*** o ***filters***, que se diferencian en la flexibilidad y la velocidad de búsqueda. Query DSL también permite realizar agregaciones de varios tipos, aunque si se desea agregar por dos o más campos es necesario anidarlas.

Elasticsearch está pensado para ser un motor de búsqueda web, y esto puede conllevar ciertas desventajas para este trabajo: el punto fuerte de Elasticsearch es su rapidez en las búsquedas, pero operaciones más complejas, como las agregaciones, tienen una sintaxis bastante incómoda y la anidación de resultados dificulta la tarea de procesarlos o agruparlos en un nuevo índice. Además, el número de resultados que devuelve por defecto es 10, y si se desean más hay que ir solicitando las páginas sucesivas. Finalmente, Elasticsearch está pensado para que cada documento sea autocontenido y por tanto resulta complicado modelar relaciones entre ellos (para crear los viajes por ejemplo).

3.3.2 MongoDB

La organización de los datos en MongoDB [16] es parecida a la de Elasticsearch: almacena la información en documentos, que se agrupan en colecciones incluidas dentro de una base de datos. Sin embargo, MongoDB guarda los datos en formato BSON (JSON codificado en binario) para que ocupen menos.

Las operaciones de filtrado o agregación se realizan sobre una colección dada, siendo siempre el resultado un nuevo conjunto de documentos sobre el cual se pueden enlazar más operaciones. El lenguaje de consultas de MongoDB incluye operadores para realizar tareas comunes, como agrupar por ciertos campos o concatenar cadenas de caracteres. También ofrece la posibilidad de realizar operaciones ***map-reduce***, en las que primero se realiza una tarea de agrupación sobre cada uno de los documentos (*map*), y después se aplica una función a cada uno de estos grupos hasta reducirlos a un solo documento (*reduce*). Estas operaciones son muy flexibles porque las funciones se escriben directamente en JavaScript, pero menos eficientes que las agregaciones.

4 Diseño y desarrollo

4.1 El conjunto de datos

Los datos de los que se ha dispuesto para la realización de este estudio son los mensajes (tweets) publicados por los usuarios en la plataforma Twitter entre el 27 de noviembre de 2012 y el 30 de noviembre de 2014, y geolocalizados en España peninsular, Baleares, Ceuta y Melilla. La geolocalización es una opción que proporciona Twitter a sus usuarios, de forma que mientras esté activada cada mensaje llevará asociadas las coordenadas (longitud y latitud) desde las que se envió el tweet. En total se dispone de **72.9 millones de tweets** que cumplan estas condiciones, publicados por **1.5 millones de usuarios distintos**, casi el cuádruple del volumen de datos utilizado en [5].

Cada tweet proporcionado por la API de Twitter contiene una gran cantidad de campos en los que se detalla toda la información existente alrededor de dicho mensaje; además de los campos básicos como pueden ser el identificador, el texto, o la fecha de publicación, se incluyen otros como por ejemplo el número de retweets y de veces marcado como favorito, el lugar asociado, si contiene imágenes o enlaces a páginas web, o el estado del usuario en el momento de publicarlo [17].

Por otra parte, se han extraído de diversas fuentes oficiales los datos demográficos de las regiones de España. A nivel de municipio, la **población total y población activa** desglosada por municipio, sexo y tramo de edad es publicada anualmente por el Instituto Nacional de Estadística (INE) [18], mientras que el dato de **número de personas desempleadas** lo proporciona el Servicio Público de Empleo Estatal (SEPE) de forma mensual, también desglosado por sexo y edad [19]; combinando estas dos fuentes de datos se puede obtener una aproximación del paro mensual a nivel de municipio. Los datos de paro de la Encuesta de Población Activa (EPA) son más precisos porque se obtienen a través de encuestas en lugar de a través del paro registrado, pero desafortunadamente estos datos se recogen a nivel de provincia como división más pequeña, y de forma trimestral en lugar de mensual, con lo cual no resultan útiles dada la amplitud limitada del conjunto de datos.

Finalmente, para ubicar los mensajes dentro de los municipios de España ha sido necesario obtener un **mapa del territorio**, sin incluir las Islas Canarias, que contenga los polígonos a nivel de municipio. Dentro de las fuentes consultadas, los datos con formas mejor definidas son los proporcionados por el Instituto Geográfico Nacional [20], aunque los nombres e identificadores de las distintas áreas administrativas se han extraído de los mapas disponibles en [21] por ser más completos y estar mejor organizados, a excepción de las comarcas, cuyos nombres han sido completados utilizando una tabla del año 1999 proveniente de la página web del INE [22]. El mapa contiene 8111 polígonos a nivel de municipio (nivel 4), aunque para algunos de ellos el INE no proporciona datos demográficos (se trata de territorios inhabitados no pertenecientes a ningún municipio oficial); de esta forma, disponemos de información completa para un total de 8027 municipios.

4.2 Estructura y funcionamiento del sistema

Uno de los objetivos del trabajo era diseñar y construir un sistema que automatice la gestión de todos los datos necesarios en el estudio. En el sistema implementado pueden identificarse tres partes: el preprocesamiento de los datos, la generación de variables, y la predicción del nivel de desempleo.

4.2.1 Preprocesamiento de los datos

La mayor parte de la información proporcionada por Twitter para cada mensaje no es necesaria en el cálculo de las variables previstas, por lo que únicamente se conservan los siguientes campos:

- Identificador de usuario
- Texto del mensaje
- Fecha y hora de creación
- Coordenadas desde las que fue enviado
- Identificadores de los usuarios mencionados en el texto

El lugar asociado al tweet se descarta porque Twitter indica en su documentación que éste no tiene por qué ser el lugar desde el que se escribió el mensaje [17]. Por tanto, se determina el municipio al que pertenece cada tweet a partir de sus coordenadas utilizando los polígonos del mapa descrito anteriormente. Se añade entonces un campo nuevo a los tweets que indica el identificador del municipio en el que está ubicado. Por una cuestión de precisión en la definición de los polígonos, algunos tweets ubicados en las costas no quedan dentro de ningún municipio; estos tweets suponen únicamente en torno al 0.1% del total del periodo estudiado y por tanto se ha decidido descartarlos. En el mapa mostrado a continuación se observa qué cantidad de los tweets del periodo estudiado han sido ubicados en cada uno de los municipios del territorio.

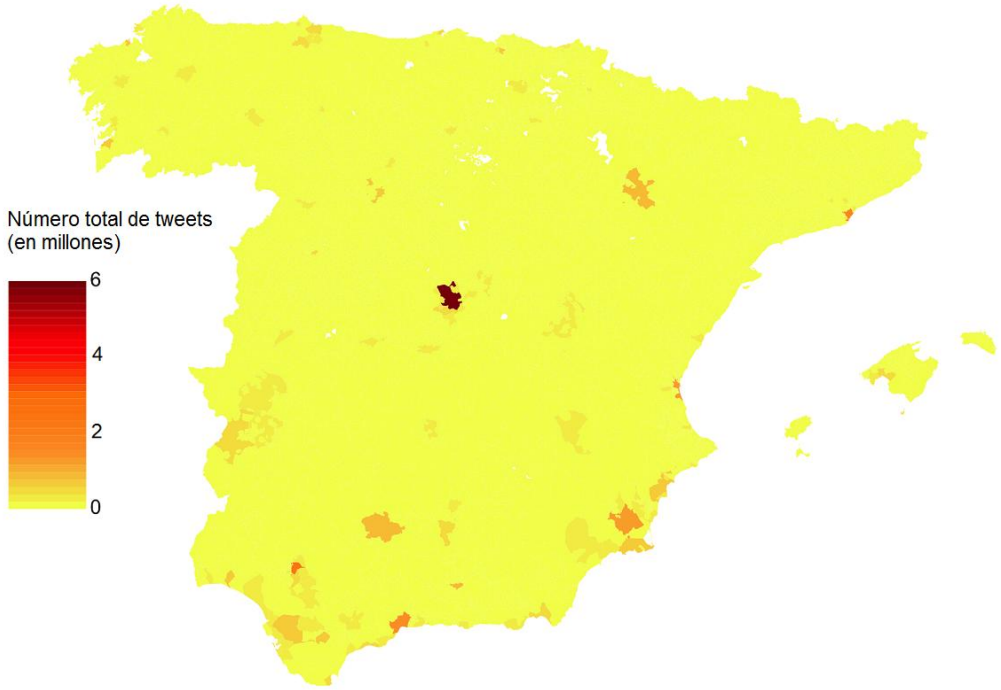

Figura 4 - Distribución de los tweets geolocalizados en España

Además de analizar la geolocalización, existen también ciertas cuestiones acerca del texto de los tweets que es necesario comprobar. Para ello, el primer paso es limpiar estos mensajes, para lo cual se han realizado las siguientes acciones en orden: eliminar las menciones, sustituir las tildes y las diéresis, quitar enlaces a páginas web, eliminar los caracteres no alfanuméricos, eliminar los espacios de sobra y, finalmente, convertir todo el texto a minúsculas. Esta limpieza resulta de gran utilidad porque, además de reducir el espacio en disco que ocupan los mensajes, facilita y agiliza el análisis de contenido que se pueda querer realizar. En las variables referentes al contenido de los tweets se usan únicamente aquellos que están escritos en castellano; en lugar de usar el campo de idioma proporcionado por Twitter, éste se determina mediante la herramienta *Chromium Compact Language Detector*, utilizada con éxito para este fin en otros estudios [23].

Una de las variables propuestas hace referencia a las faltas de ortografía en los textos. Será necesario por tanto determinar para cada tweet la presencia o no de expresiones mal escritas. Para ello se utiliza la lista de 618 faltas construida por A. Llorente *et. al.* mencionada en la sección 2.3, y se buscan estas expresiones en el texto de los tweets. Así, se añade un nuevo campo al tweet que indica si el texto contiene o no alguna falta de ortografía. No se tienen en cuenta las tildes porque su ausencia es bastante habitual cuando se escribe desde dispositivos móviles. Las faltas contempladas son tanto palabras sueltas como expresiones; en la Tabla 1 se muestran algunos ejemplos.

Falta de ortografía	Forma correcta
llendo	*yendo*
cojeremos	*cogeremos*
haber que tal	*a ver que tal*
hechas de menos	*echas de menos*

Tabla 1 - Ejemplos de faltas de ortografía contempladas

De esta forma, cuando se dispone de nuevos datos provenientes de Twitter, se procesan de la forma explicada para obtener todos los campos que se utilizarán después en el cálculo de variables. Los tweets limpios se almacenan en una tabla con los campos

id_usr	**timestamp**	**lon**	**lat**	id_munic	texto	falta_ortg	idioma

En cuanto a los datos demográficos, como están desglosados por grupos es necesario realizar un pequeño procesamiento para agregarlos. Los datos de ambos sexos se juntan, pasando así a disponer de los datos únicamente desglosados por edad. Para obtener la población total se suman todos los grupos de edad, para obtener la población activa se hace lo propio con los grupos de edad que comprenden las edades entre 15 y 65 años, y para obtener la población joven se usan solo los grupos 15-19 y 20-24 años. El número de desempleados registrado viene separado en tres grupos de edad: menores de 25, entre 25 y 44 años, y a partir de 45 años de edad; el dato total se obtiene sumando los tres grupos, mientras que el dato relativo a los desempleados jóvenes es directamente el grupo de menos de 25 años de edad. Para obtener los porcentajes de paro total y paro joven, simplemente se divide el número de desempleados entre la población activa del grupo correspondiente.

4.2.2 Generación de variables

El objetivo en esta fase es construir las variables que se utilizarán en los modelos de predicción del nivel de desempleo; por tanto, llegado este momento de la implementación, es necesario plantearse qué necesidades se van a dar en el proceso predictivo.

Las posibilidades de predicción son muy amplias y el módulo de generación de variables debe ser capaz de dar cabida a todas ellas. Por una parte, puede resultar interesante intentar predecir los cambios en el paro a lo largo del tiempo (de forma mensual, trimestral, semestral, etc.), teniendo en cuenta lo ocurrido en las redes sociales durante un periodo de tiempo determinado independiente del horizonte de predicción (en el último mes, en los dos últimos meses, en el último cuatrimestre,…). Por ejemplo, podemos desear predecir el paro de forma mensual, utilizando para ello lo ocurrido en Twitter en los tres meses anteriores al momento para el que queremos inferir el nivel de desempleo.

También puede ser interesante estudiar las diferencias en el nivel de desempleo entre distintas áreas del territorio, que no han de ser necesariamente municipios, sino también comarcas, provincias, o incluso agrupaciones de municipios definidas por el usuario.

Dado que no se espera que los cambios en el nivel de desempleo sean muy acusados en periodos de tiempo pequeños, se ha fijado el intervalo de extracción de las variables en un mes. Es decir, para cada mes se generará una matriz $n\,x\,v$, siendo n el número de **divisiones territoriales** elegidas y v el número de variables; entre las variables se incluirán también los valores del paro para dicho mes, que se usarán en el modelo como variable respuesta. Si el usuario desea realizar predicciones trimestrales, por ejemplo, en lugar de mensuales, bastará con descartar las matrices de los meses que no vaya a utilizar.

Las variables de cada matriz contemplarán lo ocurrido en las redes sociales en un intervalo temporal determinado, y consistirán en una agregación sobre tales datos. Este intervalo, el denominado **tamaño de ventana,** podrá ser definido por el usuario, ya que puede afectar notablemente a los resultados de la predicción.

Así, en la generación de variables el usuario podrá configurar cuatro parámetros: el primer y el último mes a ser considerados, el tamaño de ventana, y las divisiones territoriales a utilizar. En la Figura 5 se muestra un diagrama con un ejemplo para ilustrar la cuestión temporal, en el que las ventanas tienen un tamaño de tres meses, se usan datos de un periodo total de 10 meses y el resultado son siete matrices de variables:

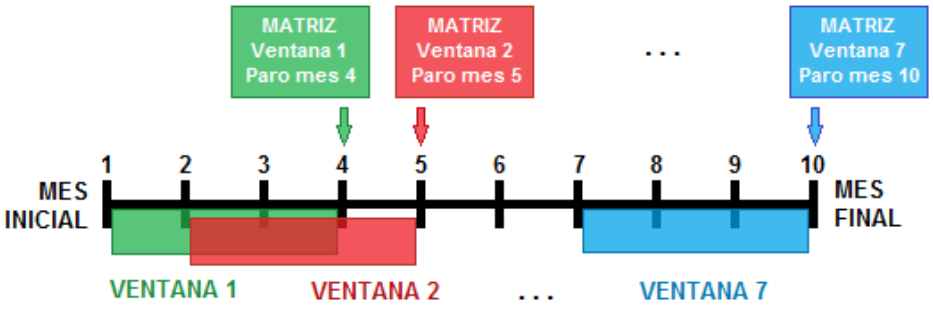

Figura 5 - Esquema temporal de la generación de variables

Dado que el usuario espera una gran flexibilidad a la hora de generar conjuntos de variables, el proceso de creación de las mismas ha de ser lo más rápido posible. Sin embargo, la cantidad de datos a procesar es muy grande y algunas variables tienen un grado de complejidad computacional notable. Por tanto, resulta imprescindible adelantar todo lo posible los cálculos, dejando para último momento únicamente aquellos que dependan de los parámetros explicados anteriormente. Dado que la agregación mínima es mensual y por municipio, la opción escogida ha sido agregar los datos originales a estos niveles y almacenarlos de esta forma como datos auxiliares para aligerar después el proceso de generación de ventanas.

Cálculos auxiliares

Pasaremos a analizar ahora los datos auxiliares que se requieren para la generación de cada una de las variables de la matriz. Téngase en cuenta que las variables definitivas se obtendrán de agregar los datos por ventana y territorio, y realizando el resto de operaciones después (fracciones, logaritmos) para obtener el resultado correcto.

Algunas de las variables necesitan disponer de la información sobre el número de hogares ubicados en cada área geográfica. Existen varias opciones en la forma de calcular esta cantidad. Recordemos que el hogar de un usuario es el lugar desde el cual ha mostrado mayor actividad, siempre y cuando suponga al menos el 40% de la actividad total, y el usuario haya publicado un número mínimo de mensajes (se trata de un umbral pequeño: 3 ó 5); si no se cumplen estas condiciones, el usuario no tendrá hogar asignado por falta de evidencia suficiente. Nótese que utilizar los datos de todo el periodo estudiado para determinar el lugar desde el que más se ha tuiteado no es adecuado si la intención es predecir el paro a futuro, puesto que estaríamos usando datos de los que no vamos a disponer en la práctica por pertenecer a un periodo de tiempo más avanzado; la única opción entonces es calcular los hogares en cada momento en función de la actividad en Twitter en los últimos tiempos. Dado que el periodo para el que se dispone de datos no es especialmente extenso, las dos posibilidades que se han contemplado son las siguientes:

- Enfoque 1: calcular el número de **hogares por cada mes**, y después agregar. Para ello, determinar para cada mes el hogar de cada usuario, usando 3 tweets como umbral para no descartarlo, y agregar estos datos para obtener el número de hogares por municipio en cada mes. Cuando se ha determinado el tamaño de ventana, obtener el número de hogares por municipio en cada periodo como la media de los valores de cada uno de los meses que se incluyen en dicho periodo.
- Enfoque 2: calcular el número de **hogares para toda la ventana**. Para ello, agregar los datos de toda la ventana (habrá de conocerse de antemano el tamaño de ventana elegido por el usuario) y determinar a partir de ellos el hogar de cada usuario, usando 5 tweets como umbral, y el número de hogares por zona.

Los datos auxiliares que requiere cada uno de los grupos de variables son los siguientes:

- **Datos demográficos**: es necesaria una tabla con los campos

mes	**id_munic**	pob_total	pob_act	desempleados	pob_act_jov	desempleados_jov

- **Nivel de utilización de Twitter**: para calcularlo se necesita el número de hogares por municipio, así que depende del enfoque elegido. Si se calculan por mes, podremos llegar a generar la tabla con los campos

mes	**id_munic**	num_hogares

Pero si se calculan por ventana, lo máximo que podemos almacenar es la tabla

mes	**id_usr**	**id_munic**	num_tweets

y determinar los hogares y calcular la distribución por municipio una vez se conozca el tamaño de ventana (en una fase posterior del proceso).

- **Actividad diaria**: considerando únicamente los tweets escritos entre semana, lo ideal en lugar de contar el número de tweets por la mañana tarde y noche con las definiciones ya fijadas, es contabilizar el número de tweets por hora, con lo que se logra una mayor flexibilidad por si se desean cambiar los intervalos horarios. Por tanto, los campos de la tabla son

mes	**id_munic**	**hora**	num_tweets

Otra opción es desagregar además por día de la semana para dar aún más flexibilidad en la creación de nuevas variables con las que experimentar.

- **Tema del texto**: la tabla auxiliar será de la forma

mes	**id_munic**	n_tw_paro	n_tw_empl	n_tw_econ	n_tw_trabajo	n_tw_total

Si en algún momento se desea añadir nuevas variables relacionadas con el contenido bastará con añadir nuevos campos a esta estructura.

- **Faltas de ortografía**: se necesita el número de personas residentes en cada municipio que han cometido faltas de ortografía, por lo que al ser necesarios los hogares también depende del enfoque. Si se calculan por mes, podremos llegar a generar la tabla con los campos

mes	**id_munic**	num_personas_con_faltas	num_hogares

Sin embargo, si se calculan por ventana, únicamente podemos precalcular la tabla

mes	**id_usr**	num_faltas

- **Interacciones**: las probabilidades de la entropía se calculan sobre la agregación de ventana y territorio, por lo que en la tabla intermedia se almacenan los campos

mes	**id_munic_A**	**id_munic_B**	num_menciones	num_viajes

Para calcular el número de menciones simplemente se miran los hogares del usuario que menciona (municipio A) y del usuario mencionado (municipio B) para todos los tweets (teniendo en cuenta que un tweet puede tener más de una mención) y después se agrega.

Para el cálculo del número de viajes, se determinan, a partir de los dos tweets que definen cada viaje, el municipio de origen (municipio A) y de destino (municipio B) y la distancia y velocidad del viaje; se eliminan los viajes de distancia menor de 1km o que son claramente erróneos por su velocidad, y el resto se agregan para crear la columna indicada.

Además de las tablas aquí descritas es necesario disponer de otra que defina las divisiones territoriales sobre las que se van a calcular las variables. Las divisiones administrativas oficiales se encuentran almacenadas en el mapa de los polígonos, mientras que para las divisiones personalizadas (por ejemplo, resultado de un algoritmo de detección de comunidades, como se explica en la sección 5.1) han de guardarse en una tabla que indique, para cada identificador de partición, qué municipios pertenecen a qué áreas:

id_particion	area	id_munic

Por ejemplo, en el caso de las particiones resultantes de algoritmos de detección de comunidades, como identificador de la partición se utilizará el nombre del algoritmo.

Generación de las matrices de variables

Por cuestiones de eficiencia, el proceso de generación de las matrices se ha dividido en dos: en el primer script el usuario indica la división territorial que desea utilizar y se realizan todas las operaciones de agregación que esta nueva información permite; en el segundo es donde el usuario define el tamaño de ventana y se generan las matrices de variables. Esto se ha hecho así para evitar tener que realizar las agregaciones territoriales cada vez que el usuario desea probar con un nuevo tamaño de ventana.

El nombre de las divisiones territoriales le permite a la herramienta identificar la partición que ha de utilizar (o bien una de las oficiales, o bien una definida por el usuario y presente en la tabla recién descrita). Identifica entonces la comunidad a la que pertenece cada municipio y agrega los datos, todos ellos contadores de diversa índole, para disponer de los mismos a nivel de la división indicada.

Como mencionamos con anterioridad, la segunda parte del proceso requiere, además del tamaño de ventana, los meses de inicio y fin y el nombre de las divisiones territoriales. Así, la herramienta sabe cuántas ventanas generar (siguiendo el esquema de la Figura 5) y qué nivel de división utilizar. Por cada una de las ventanas, se agregan los datos de los meses implicados; dispondremos así de un valor por comunidad y ventana para cada uno de los campos necesarios. En el enfoque de hogares por ventana, este es el momento en el que se calcula el hogar de cada usuario y el número de hogares por división, pudiendo entonces determinar el número de personas en cada división que cometen faltas de ortografía, disponiendo así de todos los datos auxiliares necesarios. Sólo resta, para cada ventana, operar con las columnas de datos desglosadas por comunidad para obtener las variables definitivas; la mayoría requiere únicamente dividir un campo entre otro, salvo el cálculo de las entropías que es algo más elaborado: para cada origen α, consideramos la porción de tabla siguiente:

destino	nº de interacciones
$\boldsymbol{\beta_1}$	$\boldsymbol{x_1}$
⋮	⋮
$\boldsymbol{\beta_{k_\alpha}}$	$\boldsymbol{x_{k_\alpha}}$

con $x_i \neq 0$ y $x = \sum x_i$, y donde las interacciones pueden ser viajes o menciones. Entonces,

$$H_\alpha = \frac{-\sum_{i=1}^{k_\alpha} \frac{x_i}{x} \cdot \log \frac{x_i}{x}}{\log k_\alpha}$$

Como resultado de todo el proceso obtendremos varias matrices de variables, cada una de ellas con una fila por división territorial y una columna por variable: el paro del último mes de la ventana y las once variables de Twitter medidas en toda la ventana.

4.2.3 Predicción del nivel de desempleo

La lógica de esta parte es muy sencilla: en función de los mismos parámetros de meses de inicio y final, tamaño de ventana y tipo de división, la herramienta carga las matrices correspondientes generadas en el paso previo y realiza una predicción del nivel de desempleo para cada territorio e intervalo temporal indicado (mes, trimestre, etc.). Por ejemplo, nos puede interesar predecir el desempleo por provincia para todos los meses comprendidos entre julio de 2014 y diciembre de 2014 usando una ventana de tamaño 3. La salida consistirá en un fichero con las predicciones por cada uno de los métodos de predicción que se hayan seleccionado.

Los métodos de predicción pueden ser muy variados y ofrecer resultados muy diferentes; en el apartado 5.3 se describen las pruebas realizadas a este respecto. En cuanto a la implementación, cada modelo es una caja negra que recibe una entrada y devuelve una predicción, por lo que podremos tener cuantos módulos de predicción queramos e incluso ir añadiendo más conforme se realicen nuevas pruebas con resultados aceptables.

4.3 Opciones alternativas de diseño

La tecnología que se intentó utilizar al comienzo del proyecto fue Elasticsearch; para ello, se construyó una API para R que encapsulase las operaciones CRUD del protocolo REST y se esbozó una extensión que simplificase la creación de queries en el lenguaje QueryDSL, de forma que la sintaxis particular del lenguaje fuese transparente. Sin embargo, por las desventajas mencionadas en la sección 3.3.1, el procesamiento de los datos resultaba lento y tedioso, y la inserción de los datos en la base de datos era lenta, razones por las cuales finalmente decidí pasar a utilizar en la implementación un modelo relacional.

Sin embargo, hace poco me hablaron de MongoDB y descubrí que por sus características podía ser una buena opción para el manejo de los datos del proyecto. Por tanto, aunque por limitaciones temporales ya no iba a ser posible desarrollar completamente las dos versiones del sistema (modelo relacional y modelo NoSQL) y compararlas, sí quise investigar las posibilidades de herramienta y construir y probar las queries que procesan los datos de la forma deseada, de las cuales se muestran algunos ejemplos a continuación.

Las estructuras de datos que se diseñaron para Elasticsearch son válidas también en MongoDB; existen cuatro tipos de documentos:

- **tweet**, con los campos descritos en la sección 4.2.1.
- **usuario**, con su identificador y el hogar asignado en cada mes o en cada ventana.
- **viaje**, con el identificador de usuario, los municipios y las coordenadas exactas de partida y llegada, y las fechas exactas de inicio y fin. Estos campos serán útiles para filtrar viajes poco realistas en base a la distancia recorrida y el tiempo de viaje.
- **municipio**, con su identificador, la población total, y la población activa y el número de desempleados, tanto totales como entre personas jóvenes.

Realizar agregaciones es muy sencillo; por ejemplo, para calcular el número de tweets por mes, municipio y hora de los días entre semana e insertarlos en la *collection* "ntweets_hora_diario" se puede utilizar la query mostrada a continuación. El operador `$project` selecciona o crea los campos que se desean pasar a la siguiente fase; después, el operador `$match` filtra los datos para eliminar los días de fin de semana; finalmente, `$group` agrega el número de tweets por mes, municipio y hora. Los resultados se procesan para convertir el mes en un objeto fecha (lo que facilitará después el filtrado por meses para crear las variables de la ventana) y se insertan en la *collection* "ntweets_hora_diario".

```
db.tweets.aggregate([
  {$project: {
    created_at: true,
    munic_id: true,
    weekday: {$dayOfWeek: "$created_at"}} },
  {$match: { weekday: {$gte: 2, $lte: 6 }} },
  {$group: { _
    id: {
      mes: {$dateToString: {format:"%Y-%m", date:"$created_at"}},
      munic_id: "$munic_id",
      hora: {$hour: "$created_at"} },
    ntweets: {$sum: 1} } }
]).forEach( function (doc) {
  doc._id.mes = new Date(doc._id.mes + "-01");
  db.ntweets_hora_diario.insert(doc);
});
```

Dado que recomiendan utilizar agregaciones en lugar de *map-reduce* siempre que sea posible, para crear los viajes podemos aprovechar el operador `$push`, que nos creará una lista con todos los municipios visitados por usuario y día, previa ordenación de los tweets por fecha mediante el operador `$sort`. Posteriormente, estas rutas se procesan para generar cada uno de los viajes e introducirlos en la colección "viajes".

A continuación se muestra la query que se podría utilizar para crear los viajes. Conviene guardarlos todos por si en algún momento se decide cambiar el criterio de filtrado, pero a la hora de agregarlos deberemos aplicar los filtros de distancia y velocidad comentados anteriormente (mediante los operadores `$project` y `$match`).

```
db.tweets.aggregate( [
  {$sort: { created_at: true } },
  {$group: {
    _id: {
      user_id: "$user_id",
      date: {$dateToString: {format:"%Y-%m-%d", date:"$created_at"}} },
    ruta: {$push: {
             munic_id: "$munic_id",
             when: "$created_at",
             where: "$coordenadas"}} } }
]).forEach( function (doc) {
  if (doc.ruta.length > 1) {
  for (i = 0; i < doc.ruta.length - 1; i++) {
    db.viajes.insert({
      user_id: x._id.user_id,
      munic_id_A: x.ruta[i].munic_id,
      munic_id_B: x.ruta[i+1].munic_id
      timestamp_A: x.ruta[i].when,
      timestamp _B: x.ruta[i+1].when
      localizacion_A: x.ruta[i].where,
      localizacion_B: x.ruta[i+1].where
    })
  }
})
```

5 Pruebas y resultados

5.1 Detección de comunidades socioeconómicas

A la hora de predecir el paro por territorios, se van a utilizar variables que reflejan el comportamiento de los ciudadanos en dichas zonas para inferir el paro medio de los municipios que las componen. Sin embargo, si estas divisiones son algo arbitrarias y no atienden a los patrones de movilidad reales de los habitantes, es posible que en las variables se mezclen comportamientos distintos y esta mezcla haga imposible que sirvan para dar una medida del nivel de desempleo en la zona.

Por tanto, al igual que se hace en [5], se han utilizado **algoritmos de detección de comunidades** para intentar determinar una división del territorio estudiado en comunidades socioeconómicas, de las que se espera que resulten más acertadas a la hora de identificar zonas en las que los ciudadanos tienen comportamientos similares y, por tanto, que la predicción del paro en ellas sea más precisa.

El grafo construido intenta reflejar en la medida de lo posible los patrones de movilidad de la población, y se ha formado con los datos de todo el periodo estudiado. Se trata de un grafo dirigido y ponderado, con un nodo por cada uno de los municipios que hay en España (exceptuando los de las Islas Canarias); las aristas unen nodos entre los que los usuarios de Twitter han realizado viajes, con la flecha apuntando hacia el municipio de llegada, y cuyo peso indica el número de viajes que se han realizado en esa dirección en los dos años que abarcan los datos de este estudio. En la Figura 6 se muestra un subgrafo con cuatro nodos, los cuatro municipios más poblados de España.

Figura 6 - Subgrafo de movilidad de los cuatro nodos de mayor población

Los cinco algoritmos utilizados, descritos brevemente en la sección 2.1.2, se conocen como Fast Greedy [9], Infomap [10], Label Propagation [11], Multilevel [12] y Walktrap [13]. Todos ellos funcionan para grafos ponderados, pero únicamente Infomap y Walktrap (ambos basados en caminos aleatorios) aceptan grafos dirigidos. Entre la Tabla 2 y la Tabla 3 se resumen los resultados de cada uno de los algoritmos probados.

Algoritmo	Número de comunidades	Tamaño medio comunidad	Tamaño máximo comunidad	Desv. típica tamaño
Fast Greedy	18	444	1974	510
Infomap	299	27	255	39
Label Propagation	189	42	1423	140
Multilevel	18	444	1972	500
Walktrap	387	21	444	57

Tabla 2 - Caracterización de las comunidades detectadas en el grafo de movilidad

Algoritmo	Modularidad	NMI con Com. Autónomas	NMI con Provincias	NMI con Comarcas
Fast Greedy	0.81	0.797	0.696	0.567
Infomap	0.74	0.606	0.793	0.806
Label Propagation	0.72	0.665	0.752	0.712
Multilevel	0.81	0.797	0.700	0.571
Walktrap	0.76	0.632	0.782	0.746

Tabla 3 - Medidas de calidad de las comunidades detectadas en el grafo de movilidad

Por una parte, vemos que el número de comunidades resultantes con Fast Greedy y Multilevel (Figura 7) es muy reducido, y por tanto no resultará útil si la intención es realizar una predicción con detalle dentro del territorio. Con Label Propagation (Figura 8) se obtiene un número mucho más elevado de comunidades, pero de tamaños muy dispares. Por tanto, parece que los dos algoritmos que dan un resultado más adecuado para nuestros propósitos son Infomap (Figura 9) y Walktrap (Figura 10); se trata precisamente de los dos únicos algoritmos que utilizan la dirección del grafo.

Los cinco algoritmos muestran una modularidad alta, lo que indica que todos ellos detectan una fuerte estructura en comunidades. En los dos algoritmos que obtienen pocas comunidades, éstas muestran una similitud bastante alta (casi 80%) con las Comunidades Autónomas. Las otras tres particiones tienen mayor parecido con provincias o comarcas.

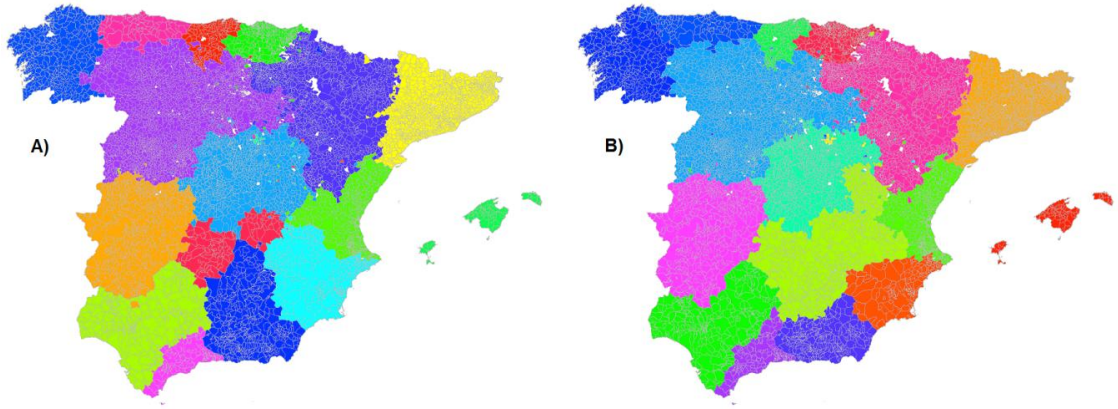

Figura 7 - Comunidades obtenidas con los algoritmos Fast Greedy (A) y Multilevel (B)

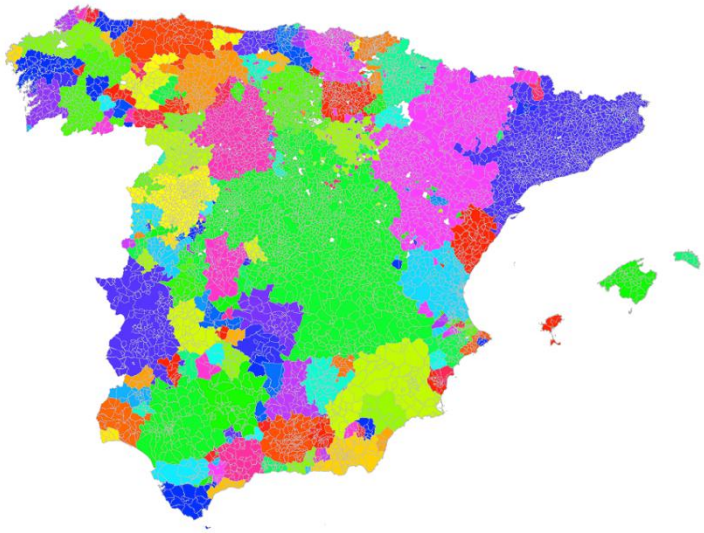

Figura 8 - Comunidades obtenidas con el algoritmo Label Propagation

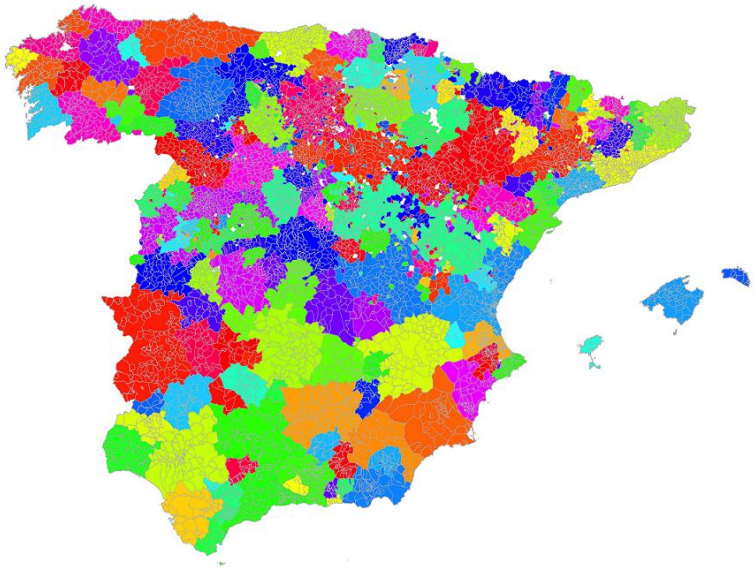

Figura 9 - Comunidades obtenidas con el algoritmo Infomap

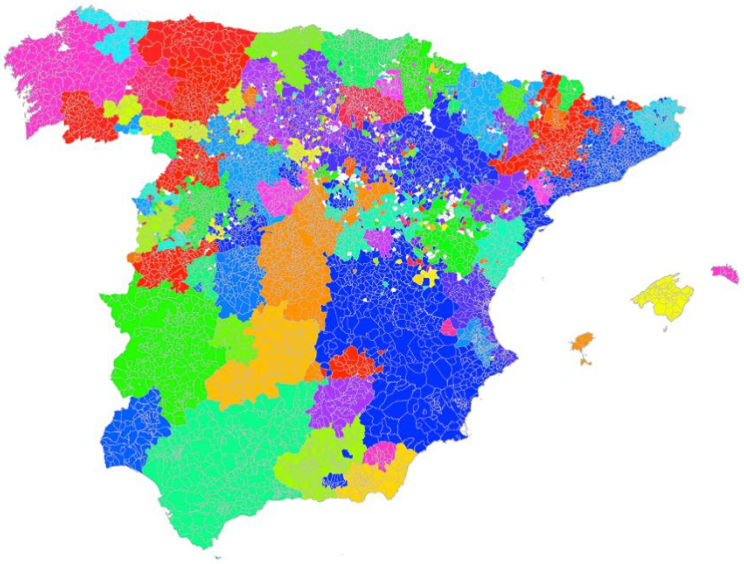

Figura 10 - Comunidades obtenidas con el algoritmo Walktrap

En las figuras anteriores se aprecian las divisiones resultantes de cada algoritmo. Se muestran en color blanco los municipios que han quedado sin asignar a ninguna comunidad. Cabe destacar que a pesar de que el grafo no contiene ninguna información acerca de la disposición geográfica de sus nodos, todas las comunidades resultantes tienen una gran cohesión geográfica. Entre las tres últimas figuras, el algoritmo que parece que proporciona divisiones más homogéneas es **Infomap**, precisamente con el que obtienen mejores resultados en [5]. Éstas serán, por tanto, las divisiones que utilizaremos como **comunidades socioeconómicas** del territorio.

Las comunidades pequeñas o con poca actividad pueden actuar como outliers que empeoren la calidad de los modelos. Para evitar esto, se consideran únicamente las comunidades con al menos 5 municipios, 1000 habitantes en total y 100 hogares localizados en ellas en todo el periodo de tiempo. En la Figura 11 se muestran las comunidades que se descartan considerando las divisiones del algoritmo Infomap; resultan aptas para el estudio un total de **131 comunidades**, que suponen el 99.3% de la población. Al realizar este mismo proceso con Walktrap, únicamente 61 comunidades pasan el filtro, las que mostraban un mayor tamaño en la Figura 10.

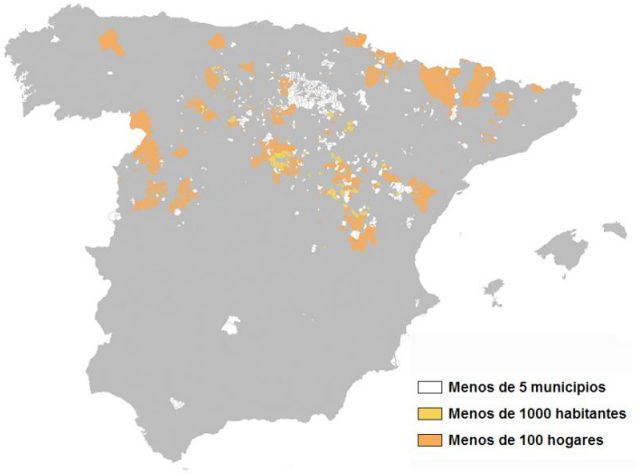

Figura 11 - Comunidades socioeconómicas descartadas para el estudio

5.2 Análisis de las variables

5.2.1 Elección del mejor enfoque

Como se ha mencionado anteriormente, existen dos aspectos en el cálculo de variables (la forma de determinar los hogares y el tamaño de ventana) para los que existen varias opciones que pueden influir en los resultados de la predicción. Por ello, se han realizado pruebas con cada una de las alternativas para intentar determinar cuál de ellas resulta más adecuada para inferir el nivel de desempleo.

Forma de calcular los hogares

Recordemos que las dos opciones que se barajaban para el cálculo de número de hogares eran bien hacer el cómputo por mes, o bien realizar el cómputo global de toda la ventana. Esto afecta a dos variables: nivel de utilización de Twitter y tasa de faltas ortográficas. Para

intentar determinar cuál de los dos enfoques es mejor calculamos para cada uno de ellos la correlación entre estas dos variables y el paro, tanto joven como total, para un tamaño de ventana fijo. El **coeficiente de correlación de Pearson** mide el grado de asociatividad lineal entre dos variables, y en este caso hará de indicador del grado de información sobre el paro que puede aportar cada variable. Los resultados se muestran en la Figura 12.

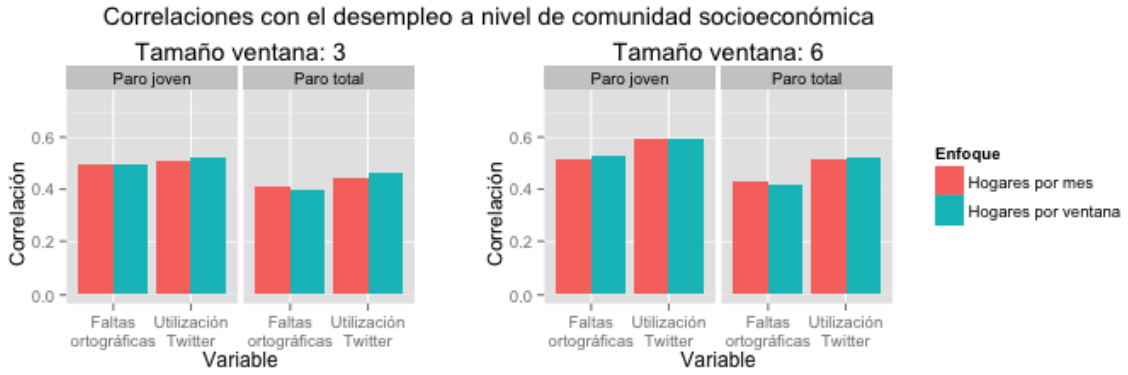

Figura 12 - Correlaciones con el desempleo según el enfoque en el cálculo de hogares

No se aprecia un claro vencedor en términos de correlación con el desempleo, puesto que las diferencias en el coeficiente son mínimas, y además no siempre está por encima el mismo enfoque; por tanto, por el momento mantenemos las dos opciones. A la vista de las correlaciones parece acertado afirmar que los resultados serán prácticamente iguales a lo largo de todo el estudio y en será difícil encontrar evidencia suficiente como para afirmar que uno de los dos enfoques es notablemente mejor.

Tamaño de ventana

Para determinar qué tamaños de ventana resultan más eficaces, se han calculado las variables con seis valores distintos para este parámetro: 1, 2, 3, 6, 9 y 12, y se ha evaluado la cantidad de información que tienen las variables sobre el paro en cada uno de los casos. Para esto, se ha realizado una regresión lineal y se ha calculado el **coeficiente de determinación R^2**, que mide qué fracción de la variabilidad del paro queda explicada por la variabilidad en las variables regresoras, en este caso los indicadores de Twitter. Para que esta medida sea independiente de la cantidad de variables incluidas en el modelo se utiliza el coeficiente de determinación ajustado R_a^2, que se muestra en la Figura 13.

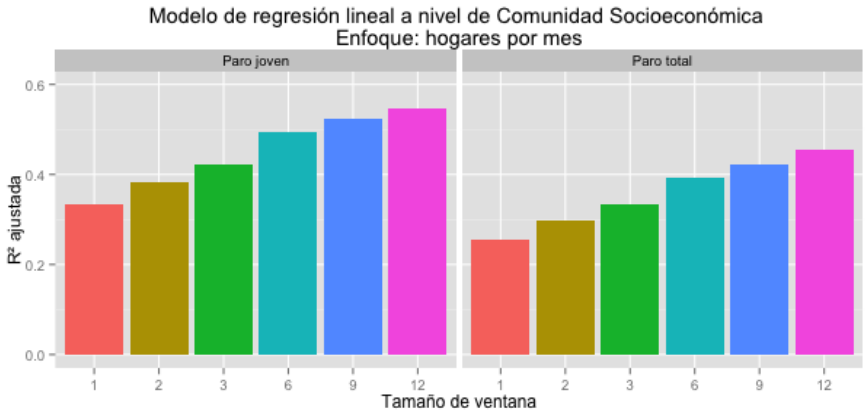

Figura 13 - Calidad del ajuste a la tasa de desempleo en función del tamaño de ventana

Se muestran los resultados obtenidos para el enfoque en el que los hogares se calculan por mes; en el enfoque de cálculo por ventana son prácticamente iguales. Se ve claramente cómo un tamaño de ventana mayor resulta más eficaz, pero parece que la mejora ofrecida por una ampliación de ventana va siendo cada vez menor a medida que el tamaño aumenta. Hay que tener en cuenta además que cuanto mayor sea el tamaño de ventana, menos matrices de variables podemos calcular, puesto que el mes de la primera ventana se retrasa y disponemos de datos para únicamente 24 meses. Por tanto, para buscar un equilibrio entre calidad de las variables y cantidad de ejemplos a proporcionar al modelo, en adelante se han usado principalmente las ventanas de 6 meses.

Por otra parte, en la Figura 13 se puede apreciar además cómo las variables de Twitter aproximan mejor el paro joven que el paro total. Esto era de esperar, puesto que el uso de las redes sociales está menos extendido entre las personas de edad elevada.

5.2.2 Valores de las variables

Los valores que toma la variable que intentamos predecir, el nivel de desempleo, a lo largo del tiempo en cada comunidad socioeconómica se muestran en la Figura 14. En general es bastante homogénea a lo largo de todo el periodo estudiado, con algunas fluctuaciones que por su forma no parecen ser debidas únicamente a la estacionalidad.

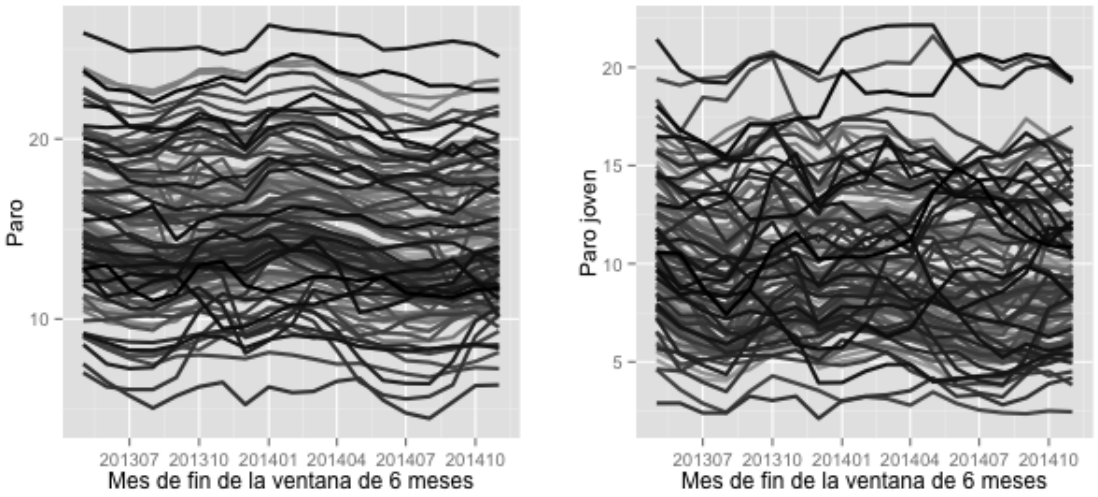

Figura 14 - Serie temporal del nivel de desempleo total y joven para cada comunidad

Para caracterizar el resto de variables, se ha estudiado el grado de asociatividad lineal que muestra cada uno de los indicadores de Twitter con el nivel de desempleo, lo cual ha ayudado a descartar algunos de ellos por la baja correlación mostrada, indicador de que la presencia de estas variables en el modelo seguramente no fuese a aportar nada. Las tres variables descartadas son: actividad por la tarde, fracción de tweets sobre trabajo y fracción de tweets sobre empleo. En la Figura 15 se muestra una comparativa entre los valores de estas variables y el paro para cada comunidad socioeconómica en cada ventana, donde el grosor de los puntos se corresponde con la población de la comunidad. En ella se aprecia cómo la correlación entre éstas y el paro es prácticamente nula; los puntos aislados que aparecen en una posición más elevada en las gráficas de tweets sobre trabajo y empleo corresponden a los valores para una misma comunidad en las distintas ventanas.

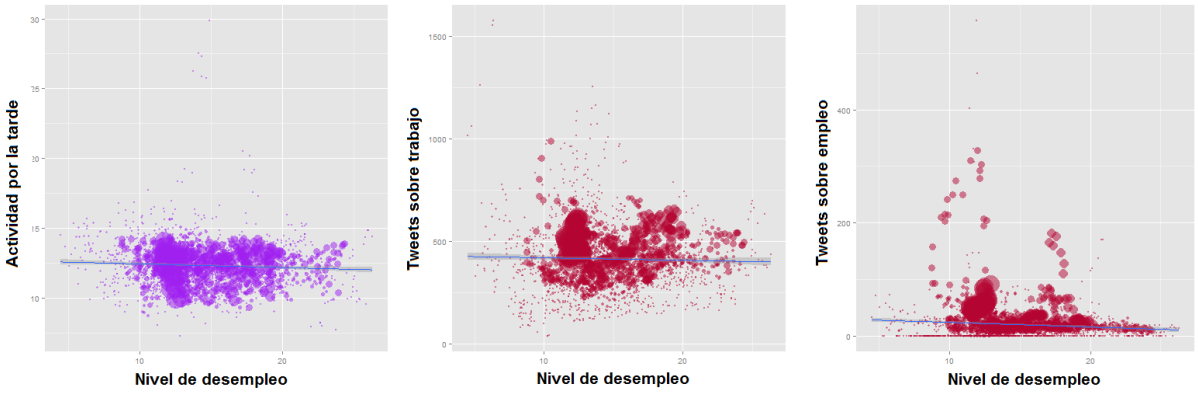

Figura 15 - Correlación entre las variables descartadas y el nivel de desempleo

A continuación se muestra un análisis más detallado de cuatro de las ocho variables que han mostrado una correlación significativa con el nivel de desempleo. El resto pueden consultarse en el Anexo (Figura 26 y Figura 27).

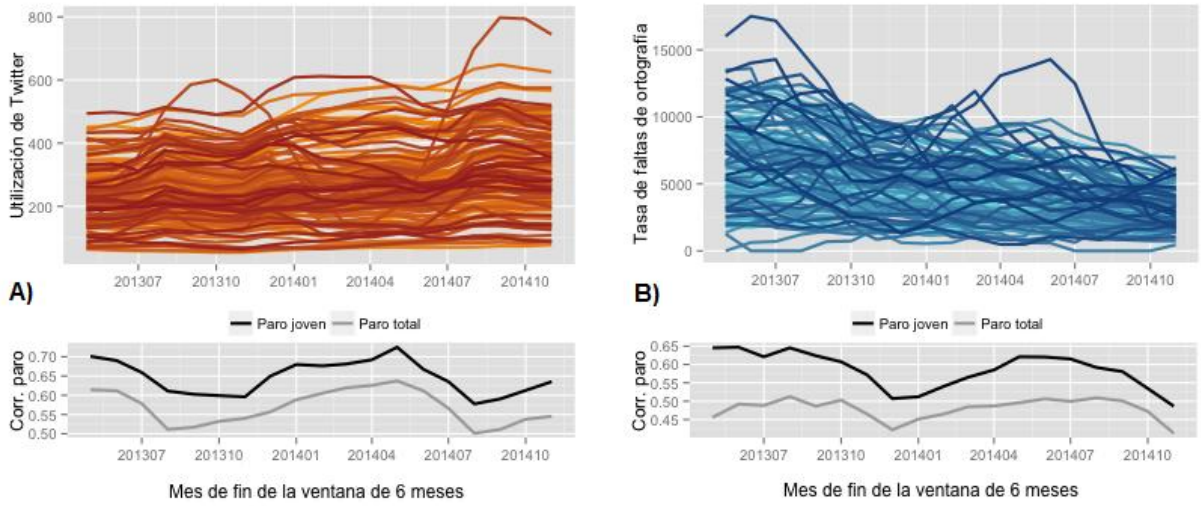

Figura 16 - Variables Utilización de Twitter (A) y Tasa de faltas de ortografía (B)

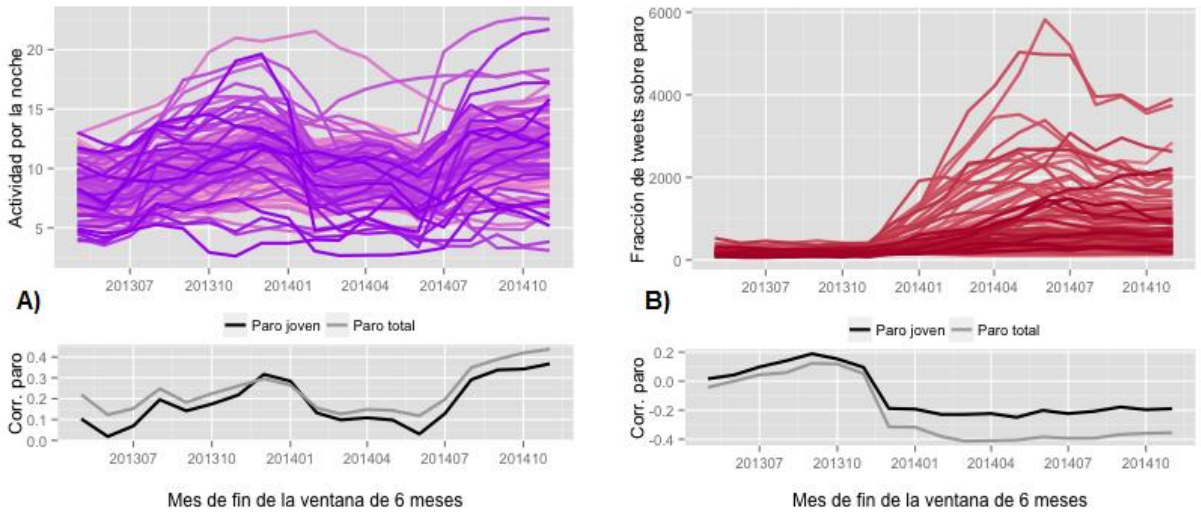

Figura 17 - Variables Actividad por la noche (A) y Fracción de tweets sobre paro (B)

Las variables más correladas con el paro son, con diferencia, el nivel de utilización de Twitter y la fracción de personas que cometen faltas de ortografía, especialmente con el paro joven. En la Figura 16.A, la correlación baja notablemente en las ventanas que incluyen los meses de verano debido al outlier que corresponde a la comunidad de Ibiza, isla con una gran afluencia de turistas en la temporada estival. Por otra parte, la actividad por la noche muestra una correlación notable con el paro en los meses de verano, pero es prácticamente nula en invierno. Finalmente, la fracción de tweets sobre "paro" es irrelevante en las primeras ventanas, pero a partir de diciembre de 2013 empieza a aumentar, y con ella, la correlación con el nivel de desempleo, especialmente el paro total.

Para el resto de análisis y en los modelos se ha eliminado la comunidad de Ibiza, puesto que es claramente una comunidad poco representativa del comportamiento en España.

5.2.3 Poder explicativo e importancia relativa de las variables

Pasamos ahora a analizar la capacidad que tienen estas ocho variables en cada ventana de tamaño 6 para explicar el paro del último mes de la ventana. Para ello se utiliza de nuevo el **coeficiente de determinación ajustado**, pero en este caso con una regresión para cada uno de los periodos. En la Figura 18 se muestran los resultados obtenidos para los dos enfoques en el cálculo de hogares, y para las tasas de desempleo total y joven.

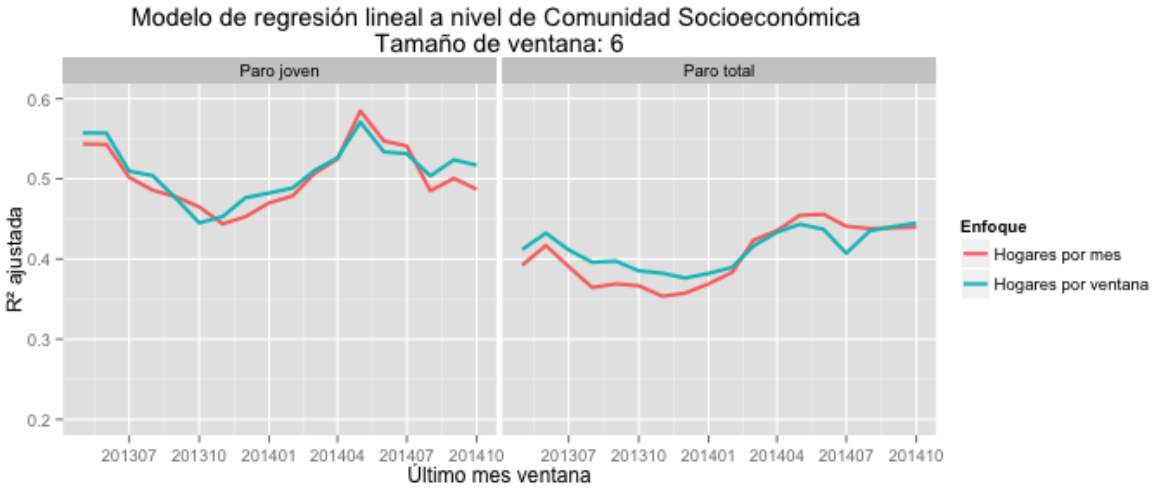

Figura 18 - Poder explicativo de los indicadores de Twitter sobre el paro

La **capacidad explicativa** es moderada, pero sorprendente si tenemos en cuenta que los indicadores están sacados exclusivamente de redes sociales, y más aún proviniendo de Twitter, una de las redes sociales con más ruido. Parece que la aproximación de hogares con ventana es algo mejor, aunque depende de la ventana y no hay gran diferencia; para simplificar el análisis a partir de ahora nos centraremos principalmente en este enfoque.

Se aprecia claramente un empeoramiento del poder explicativo en las ventanas que utilizan meses de verano. Éste es aún más bajo cuando se incluye la comunidad de Ibiza en el análisis, lo cual hace pensar que este fenómeno se debe al factor turismo en verano, que hace que cambie notablemente el comportamiento percibido en las redes sociales en las distintas zonas, mientras que la variación en el paro no es tan distinta de un área a otra. Cabe destacar que, mientras que en el paro joven parece que el R^2 vuelve a bajar en

las ventanas de verano de 2014, la capacidad de inferir el paro total se mantiene estable en el mismo periodo. Esto es probable que se deba al cambio de tendencia en la fracción de tweets sobre paro (Figura 17.B).

Al realizar este mismo análisis con ventanas de otros tamaños, se aprecia cómo un tamaño mayor suaviza el error cometido en las ventanas que contienen meses de verano (porque los meses críticos pasan a tener menos peso), aunque en los tamaños 9 y 12 la mejoría no es tan grande como la que se produce en el salto de 3 meses a 6 meses de ventana. Por otra parte, el poder explicativo de las variables en los últimos meses estudiados aumenta notablemente con los tamaños de ventana 9 y 12, tanto en el paro total como en el joven. Esto se puede consultar en la Figura 28 en el Anexo.

Resulta interesante además analizar los **pesos relativos** de las variables en el modelo de regresión lineal, que ayudan a determinar qué porcentaje de la variabilidad del paro queda explicado por cada una de las variables. El método utilizado ha sido **LMG**, propuesto por Lindeman, Merenda y Gold e implementado en el paquete *relaimpo* de R [24]; consiste en calcular el incremento de R^2 que supone añadir la variable al modelo para todos los posibles órdenes de inserción, y el promedio de esta cantidad es el peso de la variable.

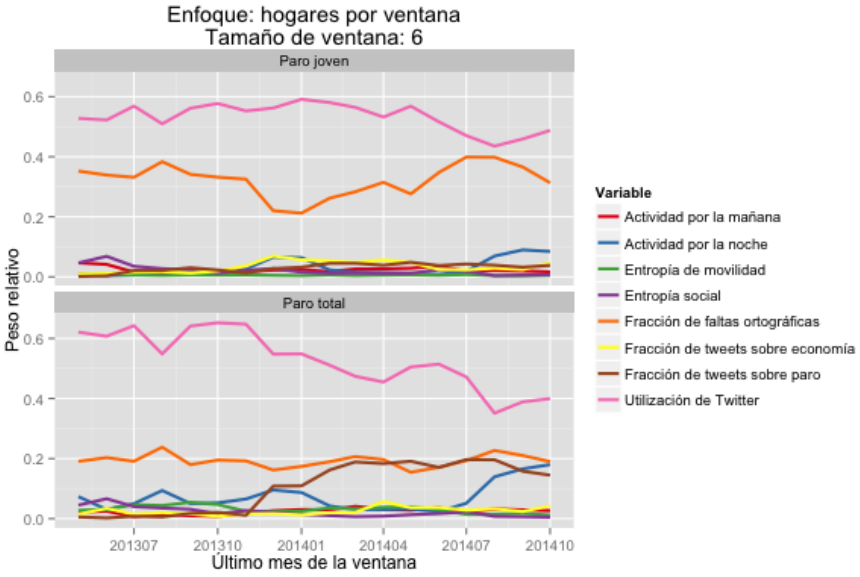

Figura 19 - Importancia relativa de las variables del modelo

La importancia de las variables difiere en función del paro a estimar: en el caso del paro joven, las dos variables con más peso son, con diferencia, la utilización de Twitter y la fracción de faltas de ortografía; la actividad por la noche parece que también influye algo en las ventanas de verano, pero el peso del resto de variables es nulo o muy puntual. Sin embargo, en la inferencia del paro total, la fracción de faltas ortográficas pasa a tener menor peso, y la importancia de la utilización de Twitter también baja porque la fracción de tweets sobre paro empieza a resultar relevante desde diciembre de 2013, cuando se produce el cambio de tendencia mencionado con anterioridad; además, el peso de la actividad por la noche es mayor, aunque también parece estacional. Para el otro enfoque de cálculo de hogares y los otros tamaños de ventana considerados, el análisis de importancia relativa de variables proporciona el mismo resultado.

En vista de los resultados, las **variables a incluir en los modelos** son las siguientes: para el paro joven, utilización de Twitter, tasa de faltas de ortografía y actividad por la noche; para el paro total se va a añadir además la fracción de tweets sobre paro.

5.2.4 Análisis de variables para otras divisiones territoriales

Se ha elegido el resultado del algoritmo Infomap aplicado al grafo de movilidad como principal división territorial a estudiar porque da lugar a un número alto, pero no excesivo, de comunidades socioeconómicas, con lo que se puede estudiar la distribución del paro con bastante detalle pero sin que los datos pierdan sentido por estar demasiado desagregados.

Aun así, resulta interesante ver qué ocurre con las **divisiones oficiales**: comunidades autónomas, provincias y comarcas. Las series temporales tienen aproximadamente la misma forma, pero las correlaciones con la tasa de desempleo cambian. La capacidad explicativa de las variables sobre el paro parece que aumenta, pero esto probablemente sea consecuencia de la disminución del número de observaciones, que hace que el modelo se ajuste mejor a los ejemplos (pero generalice peor). Sin embargo, son curiosos los resultados sobre la importancia relativa de las variables para cada una de las divisiones (Figura 20): se aprecia cómo en las divisiones pequeñas la utilización de Twitter y la tasa de faltas de ortografía tienen mucho peso; sin embargo, a medida que el tamaño de comunidad crece, van perdiendo valor y empiezan a ser más relevantes las variables que miden la actividad y, en las comunidades autónomas, las interacciones. A raíz de estos resultados parece razonable plantearse si los indicadores de movilidad empiezan a resultar representativos cuando se agregan sobre territorios amplios como las comunidades autónomas. Lamentablemente no hay datos suficientes para poder afirmarlo.

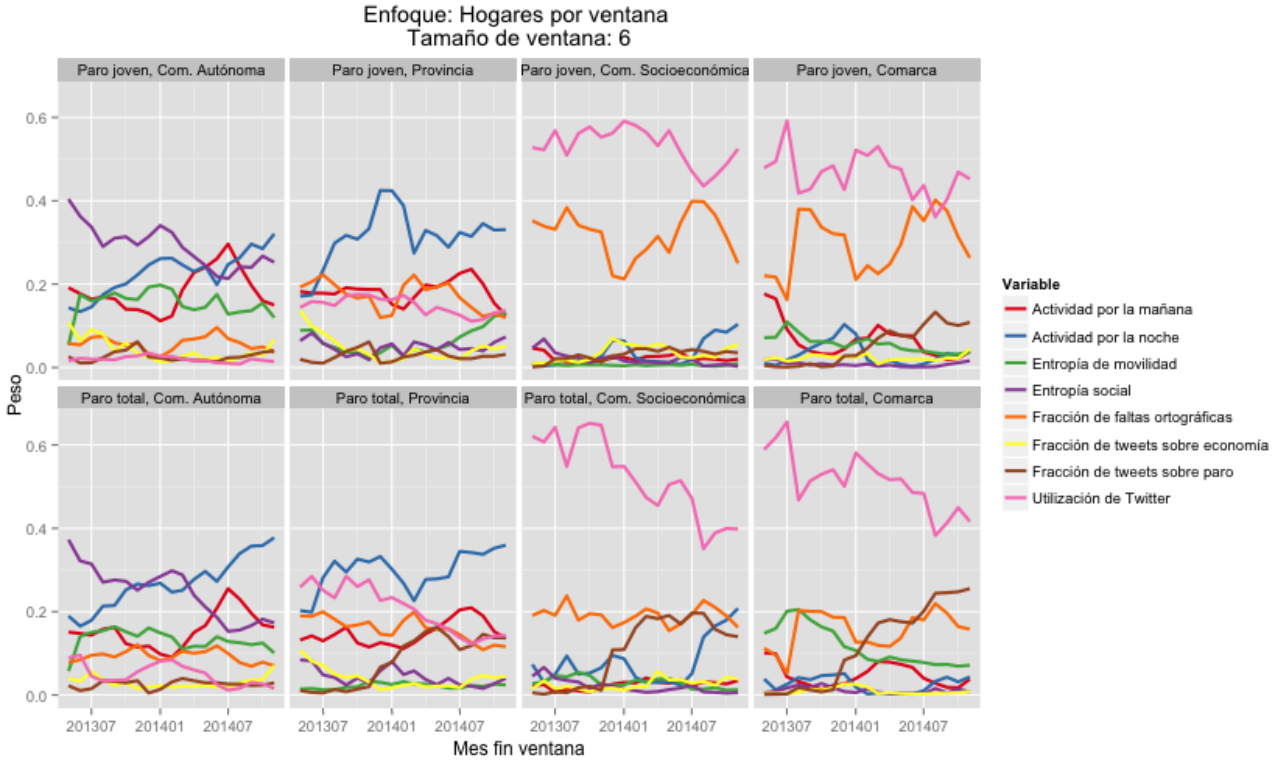

Figura 20 - Importancia relativa de las variables según la división territorial

5.3 Modelos predictivos

Pasaremos a describir ahora las diferentes pruebas realizadas para predecir la tasa de desempleo a partir de datos anteriores en el tiempo, y a exponer los resultados obtenidos.

5.3.1 Modelos para series temporales

Los modelos más extendidos para predecir el paro son los modelos autorregresivos descritos en la sección 2.2.1. La ventaja que tienen sobre los modelos de regresión lineal es que los coeficientes se optimizan para toda la serie temporal, no para un único periodo. Existen dos formas posibles de adaptar estos modelos a nuestro problema:

- Considerar cada comunidad de forma independiente y crear un modelo **ARMAX** para la serie temporal del paro de cada una de ellas, utilizando las variables de Twitter como variables exógenas que ayuden a mejorar la predicción. Después del entrenamiento obtendríamos los vectores de coeficientes ϕ y θ para los valores del paro y los errores cometidos en cada instante de tiempo anterior, respectivamente, y un vector η por cada variable para multiplicarlo por los valores que éstas toman en los periodos previos a la predicción.
- Considerar en cada instante el vector del nivel de desempleo en cada comunidad, es decir, una serie temporal multivariante, y crear un modelo **VARX** (*Vector Autoregression* con variables exógenas), con el que obtendríamos, después de entrenarlo, matrices de coeficientes en lugar de vectores.

Las pruebas con la primera aproximación han mostrado que el mejor modelo consiste en predecir simplemente lo mismo que en el instante anterior; esto se debe a que las variables del paro (total y joven) muestran tan poca variabilidad que cuando menos error se comete es cuando consideras que la serie temporal no cambia. El modelo VARX tampoco va a resultar válido entonces, puesto que al ser más complicado necesitará más datos de los que disponemos para que no haya **sobreajuste** (fenómeno que se produce cuando el modelo se adapta con demasiado detalle a los datos disponibles, y su actuación es mala con ejemplos nuevos).

5.3.2 Modelos generales adaptados a series temporales

Dado que las aproximaciones anteriores no resultan adecuadas por la limitación en el conjunto de datos, es necesario idear un método basado en un modelo más sencillo que sí pueda aprovechar los recursos de los que disponemos. Una forma de simplificar el modelo VARX sería mediante una regresión lineal que predijese el paro usando el paro del mes anterior y los indicadores de Twitter de la ventana que termina en ese mes, es decir

$$P_t = \alpha P_{t-1} + \boldsymbol{\beta} V_t$$

donde α es un escalar, P_i y $\boldsymbol{\beta}$ son vectores, y V_t es una matriz con las variables de Twitter. Pero, de nuevo, la poca variabilidad en el paro hace que ésta quede explicada directamente por la variabilidad del paro del mes anterior.

Por tanto, vamos a centrarnos en diseñar modelos que infieran el paro únicamente a partir de las variables extraídas de Twitter, con el objetivo de determinar así la potencia de estos

indicadores. Se han probado para ello varias metodologías de predicción, las cuales pasan a detallarse a continuación. Todas ellas se basan en el mismo patrón: el modelo de regresión lineal se entrena con el vector de paro de un mes como variable respuesta (un valor para cada una de las comunidades), y matriz de entrenamiento correspondiente a la ventana que acaba en ese mes, o en algún mes anterior, formada por un vector por cada una de las variables seleccionadas (consultar sección 5.2.3). Una vez se entrena el modelo, éste se puede usar para predecir el paro de un mes posterior, multiplicando los coeficientes obtenidos en el entrenamiento por la matriz que corresponda. En total se han probado alrededor de 530 modelos de estas características, todas las combinaciones posibles de tipo de tasa de desempleo (joven o total), enfoque en el cálculo de hogares (por mes o por ventana), tamaño de ventana (3, 6, 9, o 12), y otros parámetros que caracterizan al modelo y que pasaremos a describir a continuación. Se ha utilizado únicamente el algoritmo de regresión lineal porque el número de observaciones por mes no es muy alto y con algoritmos más complicados nos arriesgaríamos a tener sobreajuste.

La evaluación de los modelos se realiza midiendo el error existente entre el paro que se ha predicho $\widehat{P_i}$ y la tasa de desempleo real P_i para cada una de las n comunidades, y aplicando alguna función de agregación, generalmente el promedio. Se han utilizado dos métricas:

- **Error relativo medio**. Media del porcentaje de equivocación respecto al valor real.

$$MRE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|\widehat{P_i} - P_i\right|}{P_i}$$

- **Error absoluto medio**. Media de la diferencia con el valor real.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\widehat{P_i} - P_i\right|$$

Predicción con la ventana del mes de paro y sin reentrenar

En este método el modelo se entrena con el vector de paro P_t de un mes t y con la matriz de variables V_t de la ventana que acaba en dicho mes; a esto lo denominaremos **modelo de salto 0**. Con esto se obtiene un vector de coeficientes β_t que se utiliza para predecir el vector de paro P_{t+h} para cada uno de los meses sucesivos; para esto ha de conocerse necesariamente la matriz V_{t+h}. Por tanto, decimos que es un método en el que **no se reentrena**: los coeficientes calculados con el primer mes de ejemplo se utilizan para predecir todos los meses sucesivos.

Si el error cometido en las predicciones se mantuviese estable en el tiempo querría decir que la relación entre las variables y el nivel de desempleo es siempre la misma. Pero con el análisis de las variables de la sección anterior hemos podido comprobar que la capacidad para explicar la variable respuesta no es siempre la misma, y que las variables regresoras no tienen siempre la misma importancia. Por tanto, cabe esperar que el modelo se vaya degradando con el paso del tiempo, tal y como se muestra en la Figura 21. Cada una de las líneas de colores representa el mes con el que se ha entrenado, y se muestra el error relativo medio cometido en la predicción en cada uno de los meses sucesivos.

En general, se aprecia cómo los modelos van cometiendo un error mayor a medida que se distancia el mes de predicción del mes en el que se hizo el entrenamiento. Los modelos de color verde terminan siendo los peores, aunque no son los que más diferencia temporal tienen con el mes de predicción, porque el mes con el que han sido entrenados es uno de los que muestra un menor poder explicativo del nivel de desempleo, tal y como veíamos en la Figura 18. Además, resulta notable que los modelos entrenados para el nivel de desempleo total a partir de diciembre de 2013 se degradan mucho menos en el tiempo, lo cual puede ser debido a la ya mencionada influencia de la variable de fracción de tweets sobre paro, representada en la Figura 17.B.

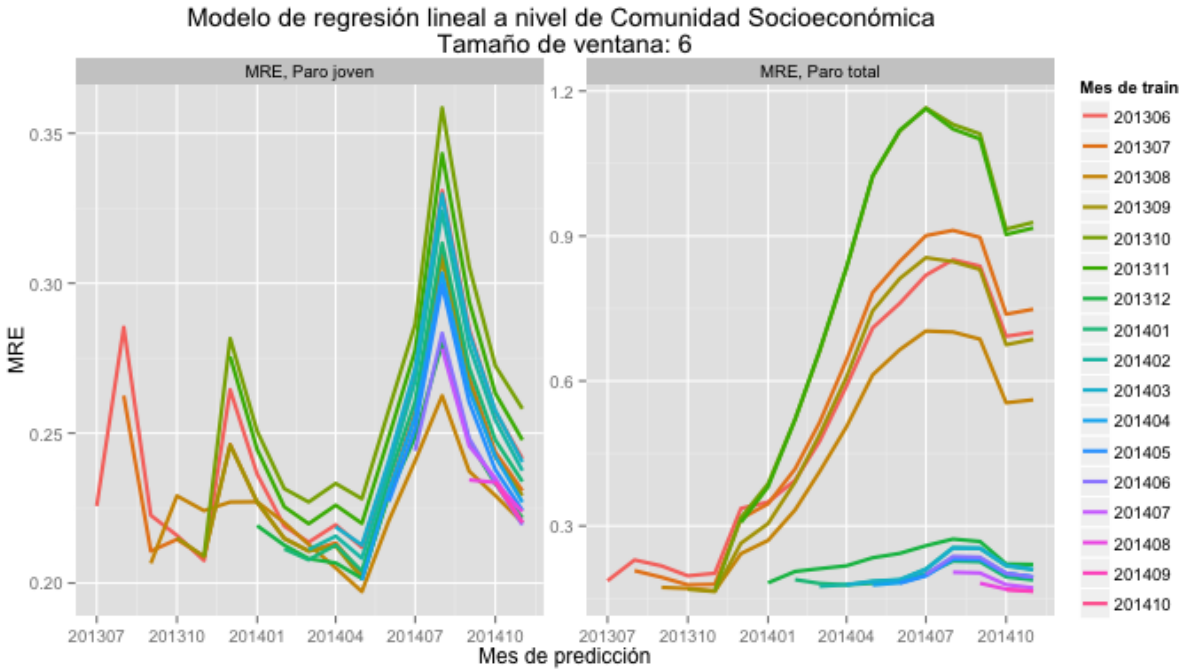

Figura 21 - Error relativo medio en el modelo de salto 0 y sin reentrenar

Predicción con la ventana del mes de paro y reentrenando

En lugar de entrenar el modelo una única vez, podemos ir calculando los coeficientes cada mes para predecir con ellos el paro del mes siguiente, esto es, realizar predicciones a **un único horizonte**. Esto debería mejorar las predicciones porque el método se va a adaptando a las condiciones a medida que pasa el tiempo.

Efectivamente, en la Figura 22 observamos que el error se mantiene más estable y disminuye notablemente en el modelo del paro total. El pico existente en el error en la predicción de 2013-12 se debe a que el modelo entrenado con los datos de 2013-11 es muy diferente, puesto que ese es el mes anterior al cambio de tendencia en la variable de tweets que mencionan el paro; por tanto, los coeficientes ajustados para este mes resultan del todo inadecuados para predecir el mes siguiente, en el que el comportamiento de las variables ha cambiado notablemente. Por otra parte, parece que el error comienza a disminuir al final del periodo, lo cual concuerda con lo que se observaba en la Figura 18, donde las variables tenían en este periodo una mayor capacidad explicativa sobre el paro total que la tendencia que se venía observando.

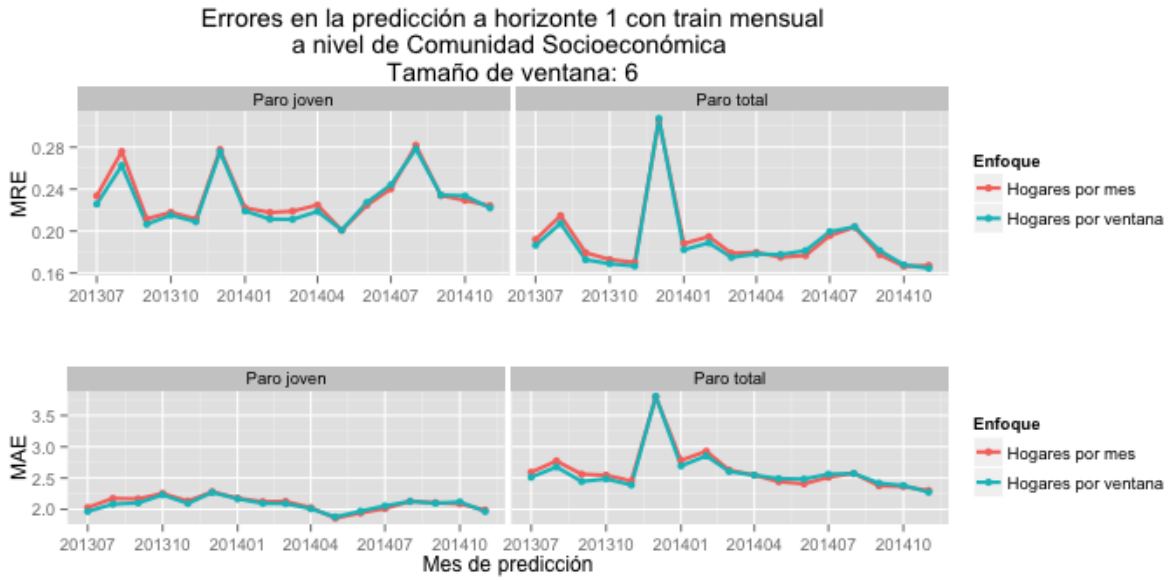

Figura 22 - Errores en el modelo de salto 0 y reentrenamiento mensual

Parece que se confirma que el enfoque de hogares por ventana responde mejor, aunque la diferencia es mínima. El error absoluto medio se mantiene estable en torno a 2 ó 2.5 puntos, dependiendo del tipo de paro. Es un resultado bastante aceptable teniendo en cuenta que el valor del paro en las distintas comunidades oscila entre los 3 y los 25 puntos.

El mismo modelo probado con ventanas de mayor tamaño produce errores algo más suavizados, pero que se mueven más o menos en la misma horquilla que en la ventana de tamaño 6. Sin embargo, con tamaño de ventana 3 los resultados son bastante peores.

Predicción con una ventana anterior al mes de paro y reentrenando

Otra posibilidad es que el paro P_t de un mes se explique mejor usando los indicadores de Twitter de una ventana V_{t-k} más lejana en el tiempo. Por tanto, otra opción es entrenar el modelo con estos dos elementos, y predecir el paro P_{t+h} a partir de la matriz V_{t+h-k}. Esto recibe el nombre de **modelo de salto k**.

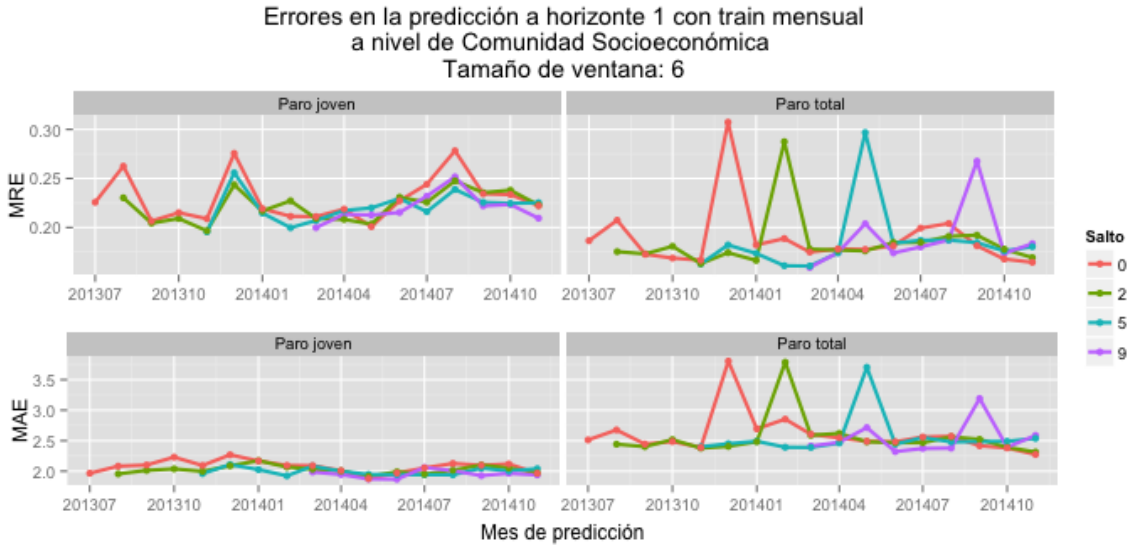

Figura 23 - Errores en modelos con saltos diferentes y reentrenamiento mensual

En la Figura 23 se muestran los errores de modelos de saltos distintos en la aproximación de hogares por ventana. La diferencia entre errores de distintos saltos es poco significativa. En el paro joven, los meses 2013-08, 2013-12 y 2014-08 siguen siendo difíciles de predecir, y en el paro total se va desplazando el pico de error, que siempre corresponde al mes que se ha predicho usando la matriz de 2013-11.

En base a todos estos análisis sobre modelos que predicen la serie temporal del paro en las comunidades socioeconómicas, concluimos que el mejor método predictivo es el **modelo de salto 0 y reentrenamiento mensual**, aunque para la tasa de desempleo total es posible que actúe ligeramente mejor el modelo de salto 2 y reentrenamiento mensual.

5.3.3 Modelos para predecir la distribución del paro en las comunidades

Hasta ahora hemos probado distintos métodos para predecir el paro en un mes a partir del entrenamiento realizado con todas las comunidades en un mes anterior; pero no hemos intentado predecir el paro en una comunidad determinada a partir de los datos del resto de comunidades. Puede ocurrir que existan varias comunidades muy influyentes que aporten mucho al modelo en el entrenamiento, y que éste empeore cuando no se usan.

Para comprobar si se está dando una situación así, probamos a predecir el paro en un conjunto aleatorio de comunidades a partir de los datos de las comunidades restantes, utilizando todos los meses disponibles. Como puede ser que este conjunto aleatorio sea especialmente bueno o malo, se ha utilizado un método de evaluación que se denomina **validación cruzada** (*cross validation)* de 10 hojas: se dividen los datos en 10 subconjuntos, y en cada uno de ellos se evalúa el modelo entrenado con los 9 grupos restantes. La métrica que evalúe el rendimiento del modelo será el promedio de las evaluaciones de cada uno de los 10 grupos.

Los modelos utilizados han sido regresión lineal y *random forest*. El número de árboles para *random forest* se ha establecido en 500, cada uno de ellos utilizando una variable, dado que únicamente cuatro variables entran al modelo. En la Figura 24 se muestran los errores obtenidos con el método de validación cruzada para los dos modelos, las dos aproximaciones en el cálculo de hogares y los dos tipos de paro. Random forest proporciona mejores resultados en todos los casos, errando un promedio de 1.75 puntos de paro. Parece entonces que el modelo se adapta bien a todas las comunidades.

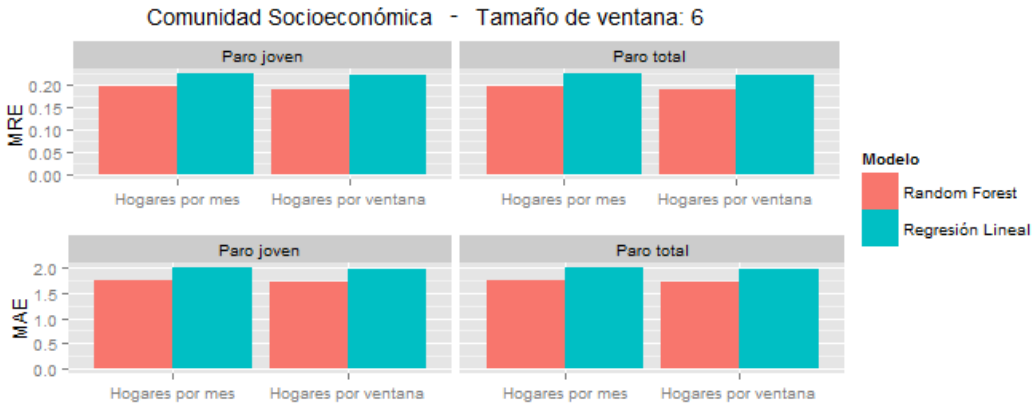

Figura 24 - Errores en la predicción del paro por comunidad socioeconómica

Una aproximación de este tipo puede resultar útil por ejemplo para inferir el paro en zonas pequeñas a partir de los indicadores de Twitter y una medida sobre el paro promedio en una zona más amplia. También puede servir para extrapolar resultados detallados de ciertas comunidades a otras que no disponen de ellos: si tenemos un número suficientemente grande de comunidades con estos datos (observaciones), podemos entrenar el modelo para que infiera los datos de otras comunidades que no se le han proporcionado en el entrenamiento.

Divisiones oficiales

Adicionalmente, se ha probado a realizar el mismo análisis para las divisiones administrativas del territorio de España (sin incluir las Islas Canarias). El número de comunidades que existen en cada división se resume en la Tabla 4.

División	Número de comunidades
Comunidad autónoma	18
Provincia	52
Comarca	319

Tabla 4 - Divisiones oficiales del territorio estudiado

De nuevo, con random forest se obtienen mejores resultados que con regresión lineal. En la Figura 25 se muestra el error cometido con el primer modelo.

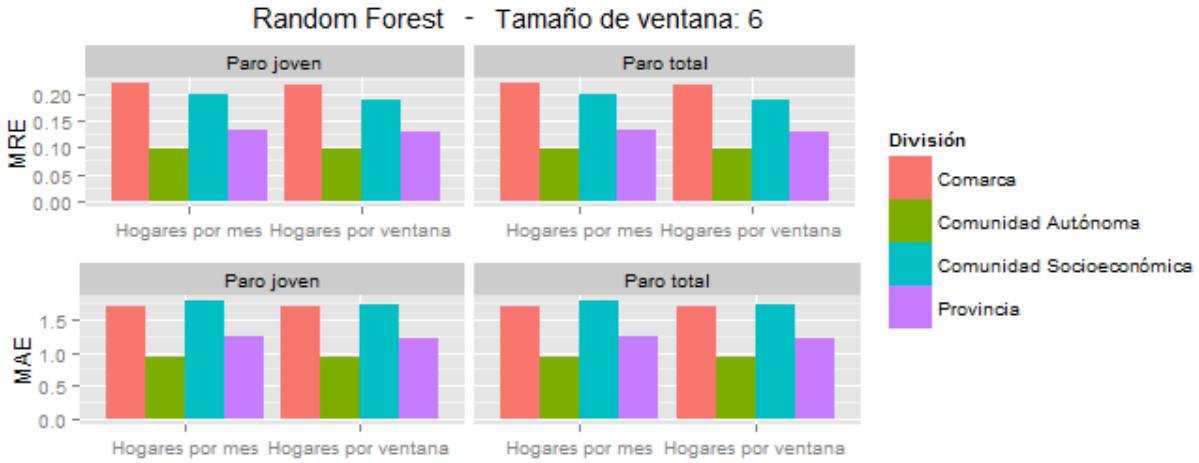

Figura 25 - Errores en la predicción del paro por comunidad para distintas divisiones

Se observa claramente cómo el error es más bajo cuanto menor es el número de comunidades en la división, llegando hasta algo menos de un punto de MAE en el caso de las comunidades autónomas. Esto es debido a que las variables en las comunidades de alto nivel son resultado de agregaciones homogéneas sobre las del nivel inmediatamente inferior, con lo cual tanto el número de datos como la variabilidad en los mismos es menor (tanto en el paro como en los indicadores sociales) y por tanto el modelo proporciona resultados más precisos.

6 Conclusiones y trabajo futuro

En este trabajo he diseñado un sistema que permite procesar los mensajes provenientes de Twitter de forma eficaz, y lo he utilizado con datos de esta red social ubicados en el territorio de España para construir indicadores que permitan inferir el nivel de desempleo en las distintas zonas geográficas. He analizado cómo de adecuadas son cada una de estas variables para esta tarea, y finalmente he construido y analizado varios modelos que predicen el paro a partir de las variables consideradas como más significativas.

6.1 Conclusiones

A continuación se enumeran las principales conclusiones obtenidas con este estudio.

- **Procesamiento eficiente de los datos**. Es crucial diseñar un sistema que procese los tweets lo más rápidamente posible porque una vez en producción los datos llegarán de forma constante y con un volumen notable (aproximadamente 100000 tweets diarios en España, pero en otros países más poblados esta cifra puede llegar a multiplicarse por 10). El contenido de los tweets hay que procesarlo al instante, y la frecuencia de las agregaciones habrá que adaptarla al volumen de datos: mensual, semanal, o incluso diaria.
- **Detección de comunidades socioeconómicas**. A partir de los patrones de movilidad de Twitter se puede dividir el territorio en zonas cohesionadas, mediante un algoritmo de detección de comunidades en grafos. Los que mejor resultados dan son los que están pensados para grafos dirigidos, porque aprovechan al máximo la información. En concreto, el algoritmo que da lugar a comunidades de tamaño más homogéneo es Infomap.
- **Variables relevantes según el tipo de paro**. A nivel de comunidad socioeconómica, los dos indicadores principales para inferir el paro joven son el nivel de utilización de Twitter y la tasa de faltas ortográficas; para el paro total resultan relevantes además la actividad por la noche y la fracción de tweets que mencionan empleo. La importancia relativa de las variables cambia notablemente para el nivel de provincia y, especialmente, el de comunidad autónoma.
- **Estacionalidad en los modelos**. Las variables tienen menor poder explicativo en verano, probablemente debido al ruido en la red que se genera con el turismo y los desplazamientos vacacionales. El error cometido en las predicciones aumenta por tanto cuando el entrenamiento se realiza con datos de meses de verano.
- **Calidad del modelo predictivo.** El mejor modelo presenta un valor de 2 para el error absoluto medio; es decir, la diferencia entre el paro real y la predicción es, en promedio, menor de 2 puntos. Éste un buen resultado teniendo en cuenta que las variables provienen exclusivamente de Twitter, pero que no compite en absoluto con

los modelos existentes para la tasa de desempleo que utilizan indicadores provenientes de otras fuentes. No obstante, existen varios aspectos que se pueden mejorar en el modelo construido, que se discuten en la siguiente sección.

6.2 Trabajo futuro

Existen varias acciones que se pueden llevar a cabo para intentar mejorar tanto el sistema como el modelo predictivo. Estas posibilidades se enumeran a continuación:

- **Comparación del rendimiento de los dos sistemas**. Es posible que el diseño con base de datos NoSQL ofrezca un mejor rendimiento, puesto que la herramienta está optimizada para manejar grandes volúmenes de datos. Por tanto, sería conveniente implementar todo el sistema utilizando este diseño y analizar qué opción es preferible a la hora de ponerlo en producción.
- **Refinamiento de los indicadores de Twitter**. Es posible que algunas de las variables calculadas tengan algo de ruido. Por una parte, el análisis de contenido llevado a cabo ha sido muy básico y puede que se hayan contabilizado tweets que en realidad no hablaban del tema deseado (por ejemplo, la palabra trabajo puede estar presente en un tweet sobre "el trabajo de biología", que claramente no interesa para nuestro análisis); por tanto, los diccionarios podrían mejorarse y extenderse (logrando así algo similar a lo que utilizan en [3]) para refinar estas variables. Por otra parte, hemos visto que algunas variables se ven negativamente influidas por los comportamientos de turistas que tuitean con geolocalización en España, por lo que otra forma de mejorar las variables podría ser detectar qué usuarios son turistas (analizando sus movimientos, o los lenguajes que utilizan en los tweets, por ejemplo) para eliminar sus mensajes de la base de datos. Es posible además que en los periodos vacacionales se produzcan errores en el cálculo de hogares debido a los desplazamientos por viajes; una forma de paliar esto, si se dispone de un periodo de tiempo suficiente, puede ser recalcular el hogar de cada usuario con los tweets de uno o dos años completos, para restarle peso al efecto de los viajes temporales.
- **Cálculo de nuevas variables.** Además de los indicadores propuestos en [5], se probaron sin éxito otros dos relacionados con la velocidad y la distancia recorridas por los usuarios. Sin embargo, es probable que exista más información que se pueda extraer de Twitter y que proporcione conocimiento adicional sobre el nivel de desempleo. El campo que más se puede explotar es el de análisis de contenido: contabilizar los tweets con nuevos conceptos o expresiones, o detectar y tener en cuenta el sentimiento del mensaje (no significará lo mismo si se menciona el paro en un texto que expresa alegría que en uno que expresa frustración).
- **Aumentar el periodo temporal del conjunto de datos.** Los modelos específicos de series temporales no han dado resultado por la limitación temporal (se suelen requerir al menos dos periodos de la serie estacionaria, mientras que para este trabajo sólo disponíamos de un periodo y medio). Aumentar el periodo de datos permitiría probar nuevas aproximaciones relativas a modelos, la posibilidad de

comprobar la periodicidad de las series para determinar la viabilidad de entrenar con datos de años anteriores, aumentar el tamaño de ventana, y validar los resultados que hemos obtenido en este estudio. Por ejemplo, el modelo para el paro total parecía que disminuía su error al final del periodo gracias al cambio de tendencia de la fracción de tweets sobre paro; disponer de los datos de meses posteriores permitiría determinar si este hecho es puntual o es una nueva tendencia. Además, un aumento notable de la serie temporal permitiría usar los datos de paro de la EPA, más precisos que los que hemos utilizado en este estudio.

- **Extender el estudio a otras áreas geográficas.** Sería interesante extender este estudio a otros países para ver si los resultados son similares. Es posible que en los países con costumbres o características socioeconómicas parecidas el mismo modelo proporcione resultados parecidos; pero en zonas totalmente distintas los indicadores más útiles pueden cambiar completamente, y la cantidad de información que éstos proporcionan sobre la tasa de desempleo, también. Si en otros países de Europa cercanos a España los resultados son similares, podría resultar interesante realizar una predicción conjunta para así poder explorar la potencia de las comunidades autónomas o las provincias, cuyo nivel de desempleo parecía inferirse a través de variables muy distintas a las relevantes en las comunidades socioeconómicas, pero que por proporcionar únicamente 16 o 50 observaciones en el territorio de España no se han podido estudiar mes a mes.
- **Probar con otros indicadores socioeconómicos**. En España se ha usado la tasa de desempleo porque es el que más cambios presenta de una comunidad a otra, pero existen otros países en los que la variabilidad es mayor en otros indicadores, como el PIB o la tasa de escolarización. Dependiendo de las características del área a estudiar puede resultar interesante intentar predecir otras métricas socioeconómicas.
- **Extraer datos de otras redes sociales**. Twitter ofrece la ventaja de que es abierta en casi su totalidad y los datos se pueden extraer con facilidad. Sin embargo, esto también hace que la información no siempre sea válida (perfiles falsos, bots, cuentas que no pertenecen a personas sino a marcas comerciales, etc). La red Flickr, por ejemplo, dispone también de una API abierta y ofrece a sus usuarios la posibilidad de etiquetar sus fotos, por lo que en países en los que su uso esté extendido podría usarse para extraer más variables, e incluso combinarlas con las obtenidas de Twitter. Esto mismo se aplica a cualquier otra red social cuyo contenido pueda ser relevante.

Bibliografía

[1] N. Eagle, M. Macy y R. Claxton, «Network Diversity and Economic Development», *Science,* nº 328, p. 1029–1031, 2010.

[2] V. Soto, V. Frías-Martínez, J. Virseda y E. Frías-Martínez, «Prediction of Socioeconomic Levels Using Cell Phone Records», de *User Modeling, Adaption and Personalization*, Springer Berlin Heidelberg, 2011, pp. 377-388.

[3] D. Antenucci, M. Cafarella, M. Levenstein, C. Ré y M. D. Shapiro, «Using Social Media to Measure Labor Market Flows», National Bureau of Economic Research, 2014.

[4] J. L. Toole, Y.-R. Lin, E. Muehlegger, D. Shoag, M. C. González y D. Lazer, «Tracking employment shocks using mobile phone data», *Journal of The Royal Society Interface,* vol. 12, nº 107, 2015.

[5] A. Llorente, M. García-Herranz, M. Cebrián y E. Moro, «Social media fingerprints of unemployment», *PLoS ONE,* vol. 10, nº 5, p. e0128692, 11 2014.

[6] W. W. Zachary, «An information flow model for conflict and fission in small groups», *Journal of Anthropological Research,* vol. 33, nº 4, pp. 452-473, 1977.

[7] S. Fortunato, «Community detection in graphs», *Physics Reports,* vol. 486, nº 3-5, pp. 75-174, 2010.

[8] M. E. J. Newman y M. Girvan, «Finding and evaluating community structure in networks», *Physical Review E,* vol. 69, nº 2, 2004.

[9] A. Clauset, M. E. J. Newman y C. Moore, «Finding community structure in very large networks», *Physical Review E,* vol. 70, nº 6, 2004.

[10] M. Rosvall y C. T. Bergstrom, «Maps of random walks on complex networks reveal community structure», *Proceedings of the National Academy of Sciences,* vol. 105, nº 4, pp. 1118-1123, 2008.

[11] U. N. Raghavan, R. Albert y S. Kumara, «Near linear time algorithm to detect community structures in large-scale networks», *Physical Review E,* vol. 76, nº 3, 2007.

[12] V. D. Blondel, J.-L. Gillaume, R. Lambiotte y E. Lefebvre, «Fast unfolding of communities in large networks», *Journal of Statistical Mechanics: Theory and Experiment,* vol. 2008, nº 10, p. P10008, 2008.

[13] P. Pons y M. Latapy, «Computing communities in large networks using random walks», de *Computer and Information Sciences - ISCIS 2005*, Springer Berlin Heidelberg, 2005, pp. 284-293.

[14] L. Danon, A. Díaz-Guilera, J. Duch y A. Arenas, «Comparing community structure identification», *Journal of Statistical Mechanics: Theory and Experiment,* vol. 2005, nº

09, p. P09008, 2005.

[15] «Elasticsearch Documentation», Elastic, [En línea]. Available: https://www.elastic.co/guide/index.html. [Último acceso: Junio 2015].

[16] «The MongoDB 3.0 Manual», MongoDB, [En línea]. Available: https://docs.mongodb.org/manual/. [Último acceso: Junio 2015].

[17] «Twitter Developers - Tweets», Twitter, [En línea]. Available: https://dev.twitter.com/overview/api/tweets. [Último acceso: Junio 2015].

[18] «INE Base - Estadística del Padrón Continuo», Instituto Nacional de Estadística, [En línea]. Available: http://www.ine.es/inebmenu/indice.htm. [Último acceso: Junio 2015].

[19] «SEPE - Paro registrado y contratos por municipios», Servicio Público de Empleo Estatal, [En línea]. Available: https://www.sepe.es/contenidos/que_es_el_sepe/estadisticas/datos_estadisticos/municipios/index.html. [Último acceso: Junio 2015].

[20] «Equipamiento Geográfico de Referencia Nacional - Líneas Límite Municipales», Centro Nacional de Información Geográfica, Instituto Geográfico Nacional, [En línea]. Available: http://centrodedescargas.cnig.es/CentroDescargas/equipamiento/lineas_limite.zip. [Último acceso: Junio 2015].

[21] «GADM database - Spain», Global Administrative Areas, [En línea]. Available: http://gadm.org/country. [Último acceso: Junio 2015].

[22] «INE Base - Metodología - Relación de Comarcas y sus Municipios», Instituto Nacional de Estadística, 1999. [En línea]. Available: http://www.ine.es/daco/daco42/agricultura/comarcas99_metodologia.xls. [Último acceso: Junio 2015].

[23] D. Mocanu, A. Baronchelli, N. Perra, B. Gonçalves, Q. Zhang y A. Vespignani, «The Twitter of Babel: Mapping World Languages through Microblogging Platforms», *PLoS ONE,* vol. 8, nº 4, p. e61981, 2013.

[24] G. Ulrike y M. Lehrkamp, «relaimpo: Relative importance of regressors in linear models», CRAN R-project, 2013. [En línea]. Available: http://cran.r-project.org/web/packages/relaimpo/index.html.

Glosario

Camino aleatorio. Trayectoria sobre un grafo que resulta de desplazarse sucesivamente de un vértice a otro de forma aleatoria con probabilidad proporcional al peso de la arista que une a los vértices.

Comunidades socioeconómicas. Comunidades resultantes del algoritmo de detección de comunidades *Infomap* sobre el grafo definido por la matriz de movilidad.

Geolocalización. Ubicación espacial (generalmente en coordenadas cartesianas).

Hogar. Para cada usuario, lugar en el que ha tuiteado con mayor frecuencia (sobrepasando cierto umbral).

Infomap. Algoritmo de detección de comunidades elegido para la definición de las comunidades socioeconómicas.

Matriz de movilidad. Matriz que recoge en sus celdas el número de viajes diarios realizados entre cada pareja de municipios.

Mención. Referencia a otro usuario en un tweet (símbolo @ seguido de su alias).

Modularidad. Métrica utilizada para determinar la calidad de una partición de un grafo en distintas comunidades.

NMI. *Normalised Mutual Information*. Métrica que determina el nivel de similitud entre dos particiones de un grafo.

Tweet. Mensaje de la red social Twitter, formado por 140 caracteres como máximo.

Variable respuesta. Variable que se desea predecir en un modelo. En este trabajo, la tasa de desempleo total o joven.

Viaje. Un usuario ha realizado un viaje entre el municipio *i* y el municipio *j* si ha tuiteado en los sitios *i* y *j* consecutivamente a lo largo del mismo día.

Anexo

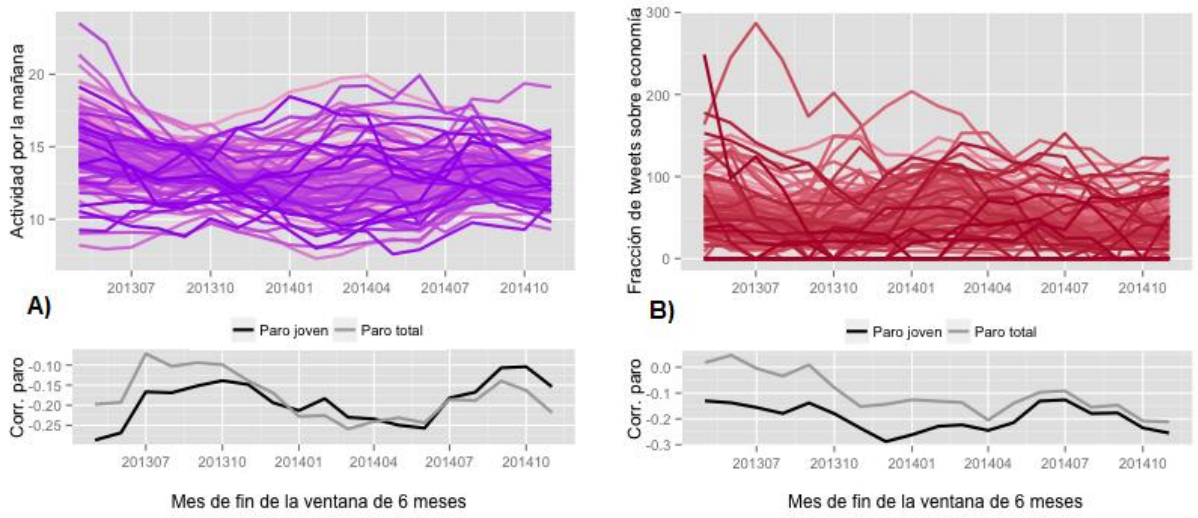

Figura 26 - Variables Actividad por la mañana (A) y Fracción de tweets de economía (B)

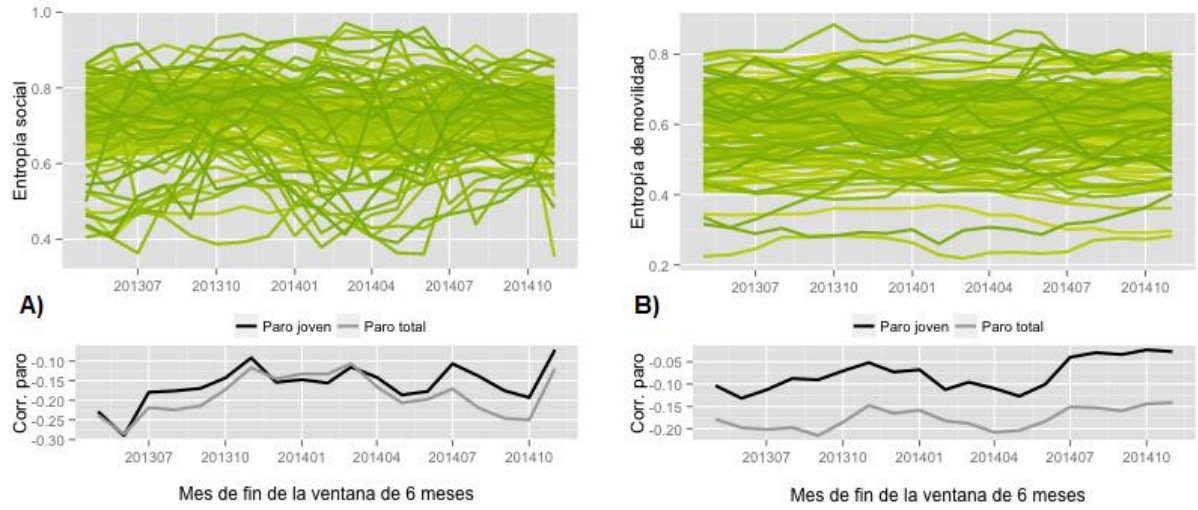

Figura 27 - Variables Entropía social (A) y Entropía de movilidad (B)

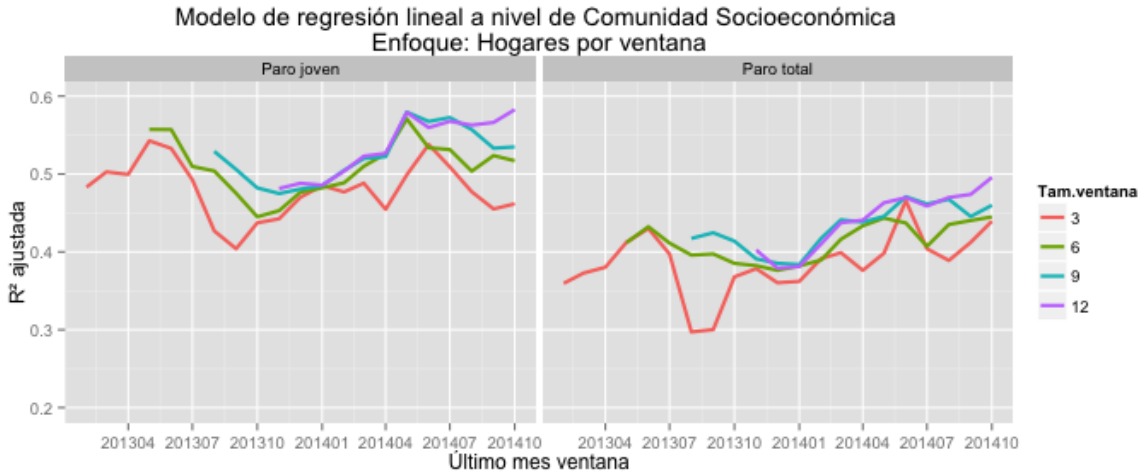

Figura 28 - Poder explicativo de las variables para distintos tamaños de ventana